(12) NACH DEM VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS (PCT) VERÖFFENTLICHTE INTERNATIONALE ANMELDUNG

(19) Weltorganisation für geistiges Eigentum
Internationales Büro

(43) Internationales Veröffentlichungsdatum
10. Mai 2001 (10.05.2001)

PCT

(10) Internationale Veröffentlichungsnummer
WO 01/33318 A1

(51) Internationale Patentklassifikation[7]: G06F 1/00

(21) Internationales Aktenzeichen: PCT/EP00/10750

(22) Internationales Anmeldedatum:
31. Oktober 2000 (31.10.2000)

(25) Einreichungssprache: Deutsch

(26) Veröffentlichungssprache: Deutsch

(30) Angaben zur Priorität:
199 53 055.6 3. November 1999 (03.11.1999) DE

(71) Anmelder und
(72) Erfinder: WITTKÖTTER, Erland [DE/CH]; Schönhaldestrasse 21, CH-8272 Ermatingen (CH).

(72) Erfinder; und
(75) Erfinder/Anmelder *(nur für US)*: SCHÜRSTEDT, Marcus, A. [DE/DE]; Kiefernstrasse 12a, 86420 Diedorf (DE).

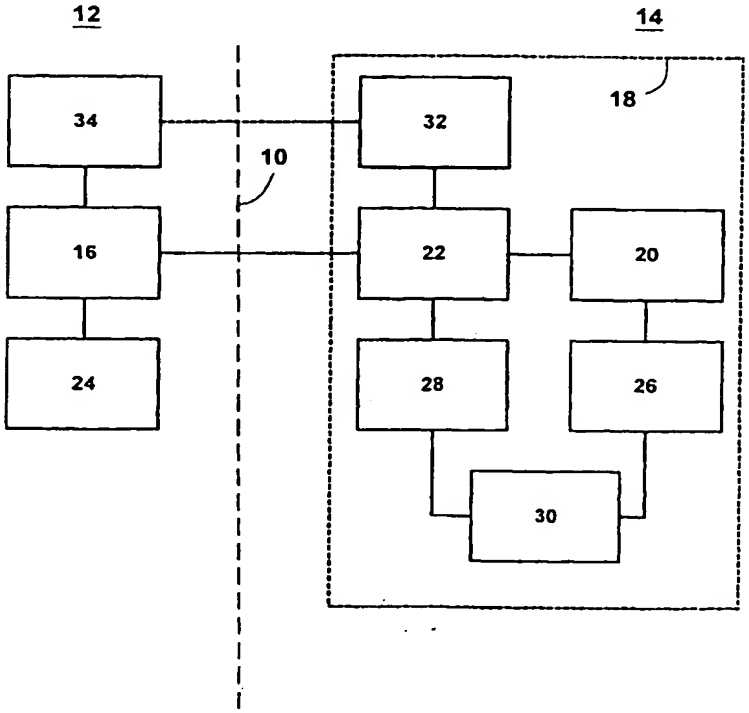
(74) Anwälte: BEHRMANN, Niels usw.; Hiebsch Peege Behrmann, Heinrich-Weber-Platz 1, 78224 Singen (DE).

(81) Bestimmungsstaaten *(national)*: IN, JP, US.

[Fortsetzung auf der nächsten Seite]

(54) Title: DEVICE AND METHOD FOR THE PROTECTED OUTPUT OF AN ELECTRONIC DOCUMENT THROUGH A DATA TRANSMISSION NETWORK

(54) Bezeichnung: VORRICHTUNG UND VERFAHREN ZUR GESCHÜTZTEN AUSGABE EINES ELEKTRONISCHEN DOKUMENTS ÜBER EIN DATENÜBERTRAGUNGSNETZ



(57) Abstract: The invention relates to a device for the protected output of an electronic document through the Internet, comprising a user-end access unit (16) and an output unit (24) which is allocated to said user-end access unit. The document received has a software instruction which can be executed for outputting material through the output unit and which contains a designation of a file or a path in the data transmission network or a form of representation of the document. Components of the document are altered using this designation, said document being prepared by a server-end deconstruction unit (26) in such a way that is only suitable for use in the form provided for the user after the software instruction has been executed. A reconfiguration unit (28) which is allocated to the server unit is configured in such a way that the software instruction or the document components can be formed in such a way that when the electronic document is received by the user again following another access operation, the instruction or document components to be loaded with the same is/are modified automatically.

(57) Zusammenfassung: Vorrichtung zur geschützten Ausgabe eines elektronischen Dokuments über das Internet, mit: einer benutzerseitigen Zugriffseinheit (16); einer der benutzerseitigen Zugriffseinheit zugeordneten Ausgabeeinheit (24), wobei das empfangene Dokument eine programmtechnische Anweisung aufweist, die für

[Fortsetzung auf der nächsten Seite]

WO 01/33318 A1

(84) **Bestimmungsstaaten** *(regional)*: europäisches Patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Veröffentlicht:

— *Mit internationalem Recherchenbericht.*

— *Vor Ablauf der für Änderungen der Ansprüche geltenden Frist. Veröffentlichung wird wiederholt, falls Änderungen eintreffen*

Zur Erklärung der Zweibuchstaben-Codes, und der anderen Abkürzungen wird auf die Erklärungen ("Guidance Notes on Codes and Abbreviations") am Anfang jeder regulären Ausgabe der PCT-Gazette verwiesen.

das Ausgeben durch die Ausgabeeinheit ausführbar ist und eine Bezeichnung einer Datei oder eines Pfades im Datenübertragungsnetz oder einer Darstellungsform des Dokuments enthält, wobei mittles der Bezeichnung Dokumentkomponenten des Dokuments geändert werden, wobei - das Dokument mittles einer serverseitigen Dekonstruktionseinheit (26) so vorbereitet ist, dass es in der für den Benutzer vorgesehenen Form erst nach dem Ausführen der programmtechnischen Anweisung brauchbar ist; und eine der Servereinheit zugeordnete Rekonfigurations-Einheit (28) so ausgebildet ist, dass die programmtechnische anweisung oder die Dokumentkomponenten so gebildet werden können, dass ein erneutes Empfangen des elektronischen Dokuments durch den Benutzer nach einem weiteren Zugriff eine Änderung der Answeisung oder eine Änderung der damit zu ladenden Dokumentkomponenten bewirkt.

BESCHREIBUNG

Vorrichtung und Verfahren zur geschützten Ausgabe eines elektronischen Dokuments über ein Datenübertragungsnetz

Die vorliegende Erfindung betrifft eine Vorrichtung zur geschützten Ausgabe eines elektronischen Dokuments über ein bevorzugt öffentliches Datenübertragungsnetz, insbesondere das Internet, mit den Merkmalen des Oberbegriffs des Patentanspruchs 1; ferner betrifft die vorliegende Erfindung ein entsprechendes Verfahren.

Im Stand der Technik wird eine derartige, gattungsgemäße Vorrichtung beispielsweise durch einen PC realisiert, mit welchem mittels bekannter Internet-Daten- und Übertragungsprotokolle auf einen Internet-Server zugegriffen werden kann, und wobei der benutzerseitige (lokale) PC zu diesem Zweck mit einer geeigneten Datenübertragungs- und Zugriffsfunktionalität (Browser) versehen ist.

Genauer gesagt findet ein bekannter Zugriff auf einen Internet-Server durch die benutzerseitige Zugriffsstation (PC) so statt, dass -- bei physikalisch hergestellter Datenverbindung zu einem geeigneten Internet-Diensteanbieter -- der Benutzer durch Eingabe einer zugehörigen, individualisierenden Adresse mit dem von ihm gewünschten Server Kontakt aufnehmen kann und dann ein Datenkommunikationsvorgang in durch bekannte Internetprotokolle geregelter Weise im wesentlichen dadurch erfolgt, dass der Benutzer lokal vom Server angebotene elektronische Dokumente empfangt, mittels seiner Zugriffseinheit aufbereitet und darstellt, und es ihm sein lokaler PC zudem ermöglicht, zugehörige Zugriffsbefehle, Dateinamen usw. nach entsprechender (Tastatur-) Eingabe oder Bezeichnung mit einem Zeigegerät (Maus) der netzseitigen Servereinheit zu übermitteln.

In der aktuellen Datenkommunikation über das Internet hat sich dabei insbesondere das Übertragen von elektronischen Dokumenten in Form einer (Bildschirm-) Seite im HTML-Format durchgesetzt, wobei durch die lokale Ausgabeeinheit, etwa der geeignet konfigurierte Browser, dann das HTML-Dokument mit den zugehörigen Inhalts- und Formatkomponenten in der für den Benutzer vorgesehenen Weise (d. h. in der gewünschten Form, dem gewünschten Erscheinungsbild und mit dem vorgesehenen Inhalt) vom Benutzer auf den PC-Bildschirm betrachtet oder von einem angeschlossenen Drucker ausgedruckt werden kann.

Insbesondere bei komplexeren elektronischen Dokumenten (speziell solchen, bei welchen ein auf einem Bildschirm darzustellender Text zusätzlich Fotos oder andere graphische Elemente aufweisen kann) enthält jedoch das benutzerseitig beim Erstkontakt mit dem Server geladene HTML-Dokument nicht unmittelbar die zugehörigen Bild- oder Graphikdaten, vielmehr ist in dem elektronischen HTML-Dokument selbst wiederum eine Pfad- und Dateiangabe enthalten, mit welcher die benutzerseitige Zugriffseinheit eine zugehörige Foto- oder Graphikdatei vom Server laden kann und die Ausgabeeinheit dann die entsprechenden Bild- oder Graphikdaten in das darzustellende, komplexe elektronische Dokument einfügt.

Insoweit besteht also eine herkömmliche HTML-Dokumentseite nicht nur aus (unmittelbar von der Ausgabeeinheit in eine Darstellung umsetzbaren) Inhalts- und Strukturkomponenten (also beispielsweise Text und dessen vorgesehener Formatierung), auch ist die beschriebene, zusätzlich zu ladende Foto- oder Graphikdatei als zusätzliche programmtechnische Anweisung zu verstehen, nämlich als Anweisung an die benutzerseitige Zugriffseinheit, die bezeichnete, auf dem Server unter dem angegebenen Namen sowie dem angegebenen Pfad auffindbare Datei in die benutzerseitige Ausgabeeinheit hineinzuladen.

In der konkreten praktischen Realisierung besteht somit das Herstellen und Anzeigen eines komplexen elektronischen Dokuments durch eine Ausgabeeinheit aus einer Mehrzahl von Zugriffen auf die Servereinheit über das Internet, bis sämtliche Inhalte des

elektronischen Dokuments (Text, Formatierungen, Bilder, Graphiken, insbesondere auch bewegte Bilder) vollständig geladen und dem Benutzer angezeigt werden können.

Zur weiteren Erhöhung der Flexibilität und der Vielfalt der Möglichkeiten in der Erstellung und Übertragung der im vorliegenden Beispiel auch als Webseiten bezeichneten elektronischen Dokumente ist es zudem bekannt, nicht nur Bild- und Graphikkomponenten einer Webseite in Form zusätzlicher Dateien vom Server heranzuführen, auch Textbausteine und weitere Bestandteile des elektronischen Dokuments werden nicht allein als HTML-Dokument mit dem Server-Erstkontakt übertragen, sondern über einen speziellen Ladebefehl bezeichnet und als Datei separat geladen. Zur Realisierung derartiger, komplexerer elektronischer Dokumente werden sog. Scriptsprachen (eine besonders gängige ist Javascript) bzw. Visual Basic-Script verwendet, in der Art einer Interpretersprache strukturierte Anweisungssequenzen, die von der benutzerseitigen Zugriffs- bzw. Ausgabeeinheit sequentiell abgearbeitet werden.

Um insbesondere bei wiederholten Zugriffen auf dieselbe serverseitige Datei im Rahmen einer Benutzersession (d. h. einer bestehenden, fortdauernden Verbindung zwischen benutzerseitiger Zugriffseinheit und Servereinheit) nicht erneut die identischen elektronischen Daten über das Internet heranführen zu müssen, ist üblicherweise im Rahmen der benutzerseitigen Zugriffseinheit eine Pufferspeichereinheit (Cache) vorgesehen, die zuletzt geladene Dateien bzw. Dateiinhalte zwischenspeichert und bei erneutem Aufrufen lokal (und damit äußerst schnell) zur Verfügung stellt; oftmals zeitraubendes Neuladen von Server-Daten, die bereits einmal geladen worden sind, wird somit unnotig.

Vor dem Hintergrund einer verbesserten Zugriffskontrolle auf potentiell schutzbedürftige Inhalte eines serverseitig angebotenen elektronischen Dokuments (als "Schutz" im Rahmen der vorliegenden Erfindung soll insbesondere auch die Möglichkeit durch einen den Server betreibenden Anbieter verstanden werden, Zugriff auf

und Umgang mit dem elektronischen Dokument in zeitlicher, örtlicher, personeller, funktioneller oder betriebssystem-/plattform-technischer Hinsicht zu kontrollieren und beispielsweise Funktionen wie Kopieren, Speichern oder Drucken des elektronischen Dokuments auf Benutzerseite nach dem Laden zu verhindern) erweisen sich solche bekannten Vorgehensweisen oftmals als unzureichend, und insbesondere die bekannte Cache-Technologie sorgt dafür, dass bereits nach einmaligem, erfolgreichem Zugriff ein Benutzer das erhaltene elektronische Dokument praktisch unbegrenzt kopieren (und damit vervielfältigen) bzw. über den Inhalt frei verfügen kann.

Die bekannten Vorgehensweisen bei der Übertragung elektronischer Dokumentdaten in der vorbeschriebenen Weise eignet sich somit nur sehr begrenzt für das kontrollierte Übertragen und Ausgeben eines elektronischen Dokuments; hierzu kommt die Möglichkeit, dass praktisch alle aktuell gängigen Browser einem Benutzer die Möglichkeit geben, mit vergleichsweise geringem Aufwand die dem Seitenaufbau eines Dokuments (bzw. einer Dokumentseite) zugrundeliegenden HTML- bzw. Javascript-Befehle darzustellen und so nicht nur die unmittelbaren Inhaltskomponenten zu erhalten, sondern auch die Pfad- und Dateibezeichnungen für serverseitig noch heranzuführende zusätzliche Dateien im Rahmen des Dokuments.

Aufgabe der vorliegenden Erfindung ist es daher, gattungsgemäße Vorrichtungen, Verfahren und Systeme zur Übertragung und Ausgabe von elektronischen Dokumenten über öffentliche Datenübertragungsnetze, insbesondere das Internet, hinsichtlich der Möglichkeiten zur wirksameren Benutzungs- und Zugriffskontrolle eines Benutzers zu verbessern und insbesondere die Möglichkeit zu schaffen, mehrfache Zugriffe auf ein elektronisches Dokument und/oder benutzerseitige Operationen wie das Kopieren, Speichern oder Drucken eines elektronischen Dokuments von gesonderten Identifikations-, Authorisierungs- und/oder Kompensationsvorgängen abhängig zu machen.

Die Aufgabe wird durch die Vorrichtung mit den Merkmalen des Anspruchs 1 sowie das Verfahren mit den Merkmalen des Anspruchs 14 gelöst; vorteilhafte Weiterbildungen der Erfindung sind in den Unteransprüchen beschrieben.

In erfindungsgemäß vorteilhafter Weise wird durch die vorliegende Erfindung erreicht, dass beispielsweise nach einmaligem, erfolgreichem Zugriff eines Benutzer auf ein von einem Internet-Server angebotenes elektronisches Dokument weitere Zugriffe und Zugriffsversuche kontrolliert werden können, da, gemäß einer bevorzugten Ausführungsform, bereits nach einem erstmaligen Zugreifen auf etwa eine mittels eines Javascript-Befehls bezeichnete Server-Dokumentdatei sich der Dateiname dieser Dokumentdatei ändert und somit ein erneuter Ladezugriff (wie er beispielsweise für ein benutzerseitiges Abspeichern oder Drucken des elektronischen Dokuments üblicherweise notwendig ist) zu einem Fehlversuch führt, wenn nicht eine spezielle Vereinbarung zwischen Benutzer und serverseitigem Anbieter getroffen wurde, mittels welcher dem Benutzer auch unter der modifizierten Adresse sinnvoller und nutzbarer Dateiinhalt zur Verfügung gestellt wird.

Da insbesondere auch der benutzerseitige Cache-Speicher etwa eine zwischengespeicherte Bilddatei unter ihrer alten, nach dem weiteren Zugriff nunmehr ungültigen Dateibezeichnung abgelegt hat, ist auch mittels des Cache keine problemlose Rekonstruktion des ursprünglich empfangenen elektronischen Dokuments möglich: Da durch die jetzt ungültige Dateibezeichnung die (zwar nach wie vor im Cache vorhandene) Bilddatei nicht mehr aufgerufen werden kann, ist der Aufbau des Gesamtdokuments unmöglich.

Im Rahmen der vorliegenden Erfindung ist dabei das "elektronische Dokument" nicht auf eine mittels gängiger Browser-Systeme darstellbare Webseite beschränkt; vielmehr ist als elektronisches Dokument im Rahmen der vorliegenden Erfindung jegliche Text-, Bild-, Video-, Musik-, Spiel-, Programm- oder Multimediadatei zu verstehen, welche über ein elektronisches Datenübertragungsnetz, beispielsweise das Internet, übertragen

werden und benutzerseitig verwendet werden kann. Dabei ist als für den Benutzer "vorgesehene Form" jeglicher Inhalt, jegliche Gestaltung und/oder jegliche Funktionalität eines vordefinierten elektronischen Dokuments zu verstehen, welche vom Anbieter für einen (authorisierten) Benutzer vorgesehen sind.

Entsprechend wirkt die "Dekonstruktionseinheit" im Rahmen der vorliegenden Erfindung so, dass das elektronische Dokument gegenüber der für den Benutzer vorgesehenen Form in seinen Eigenschaften verändert, üblicherweise damit in seiner Brauchbarkeit -- völlig -- vermindert und damit entwertet wird. Nach dieser Entstrukturierung besteht somit aus Anbietersicht des elektronischen Dokuments keine Gefahr, dass ein Kopieren, Speichern oder Drucken dieses entstrukturierten Dokuments zu einem Schaden führt. Erst durch das Ausführen der zugehörigen programmtechnischen Anweisung durch die lokale Ausgabeeinheit wird nämlich das elektronische Dokument in seiner brauchbaren, vorgesehenen Form wieder hergestellt (im weiteren auch Rekonstruktion genannt), wobei, wie am Beispiel beschrieben, geeignete Rekonstruktionsanweisungen durch die programmtechnische Anweisung (etwa über eine damit bezeichnete Serverdatei) selbst dargestellt oder herangeführt werden können.

Die ebenfalls im Rahmen der Erfindung vorgesehene, serverseitige Rekonfigurationseinheit sorgt dafür, dass die zur Wiederherstellung eines brauchbaren elektronischen Dokuments benötigten programmtechnischen Anweisungen dynamisiert werden, d. h. sich etwa bei jedem Zugriff (oder bei einer vorbestimmten Maximalzahl von Zugriffen) ändern, so dass jeder weitergehende Zugriffsversuch dann nicht zur gewünschten Rekonfigurationsdatei bzw. -Anweisung führt, mithin also das elektronische Dokument benutzerseitig nicht in der vorgesehenen Weise darstellbar ist.

Im Ergebnis hat somit die Erfindung eine äußerst wirksame Zugriffskontrolle auf elektronische Dokumente erreicht, die selbst durch aufwendige Cache-Systeme auf Benutzerseite nicht überwunden werden kann.

Weiterbildungsgemäß ist es besonders bevorzugt, die programmtechnische Anweisung im Rahmen der vorliegenden Erfindung mittels einer Scriptsprache zu realisieren, die aus der Gruppe bestehend aus Javascript, Visual Basic Script, XML, XSL und HTML od. dgl. ausgewählt ist. Insbesondere eine Kombination von Javascript-Befehlen, die XML-Dateien aufrufen, ermöglicht ein einfach zu realisierendes und trotzdem komplexes Zugriffs-Sicherungssystem für die severseitigen elektronischen Daten.

Weiterbildungsgemäß ist es zudem besonders bevorzugt, eine Mehrzahl von programmtechnischen Anweisungen zur Ausführung durch die lokale Ausgabeeinheit nicht lokal zu laden (und damit einer lokalen Speicherung oder Pufferung zugängig zu machen) sondern vielmehr auch diese programmtechnischen Anweisungen lediglich serverseitig zur schrittweisen Ausführung bereitzustellen. Durch diese Maßnahme wird die Komplexität und mithin die Sicherheit des erreichten Dokumentschutzes weiter verbessert.

Besonders bevorzugt ist es zudem, eine benutzerseitige Rekonstruktion (d. h. eine Ausführung einer programmtechnischen Anweisung zum Herstellen eines elektronischen Dokuments in der vorgesehenen Form) als Reaktion auf eine manuelle Interaktion mit dem Benutzer, z. B. Tastatureingabe, Mausbetätigen od. dgl., vorzusehen, oder mit diesen bevorzugt manuellen Aktionen die im Rahmen der Erfindung vorgesehene Änderung oder Neubildung der programmtechnischen Anweisung zu triggern. Auf diese Weise kann etwa sichergestellt werden, dass ein elektronisches Dokument lediglich zum aktuellen Zeitpunkt des Betrachtens durch einen Benutzer (und dann bevorzugt auch nur im aktuell sichtbaren Ausschnitt) in der vorgesehenen Weise zusammengefügt (rekonstruiert) ist, während ansonsten die Dokumentkomponenten lokal in der unbrauchbaren Form verbleiben.

Ein möglicher Anwendungszweck der vorliegenden Erfindung liegt darin, spezielle wertbestimmende Funktionen oder Operationen eines elektronischen Dokuments, beispielsweise das Drucken oder lokale Abspeichern, nur solchen Benutzern zu gestatten, die vor einem Zugriff auf den Server identifiziert bzw. authentifiziert

wurden und/oder mit dem Anbieter in einen Abrechnungsdialog eingetreten sind (im Rahmen dessen z. B. Abrechnungsdaten in Form einer Kreditkartennummer od. dgl. übertragen wurden). Als Reaktion auf einen derartigen Identifikations- und Abrechnungsdialog zwischen den zugehörigen, benutzer- und serverseitig vorgesehenen Einheiten würde dann die Rekonfigurationseinheit, auch bei fortlaufend geänderten programmtechnischen Anweisungen, den weiteren, ordnungsgemäßen (brauchbaren) Zugriff durch einen so autorisierten Benutzer ermöglichen, oder es würde die Rekonfigurationseinheit als Reaktion auf eine geeignete Identifikations- und/oder Transaktionshandlung deaktiviert werden, so dass während der betreffenden Benutzersession keine Neubildung oder Änderung erfolgt.

Besonders bevorzugt ist es im Rahmen der Erfindung, zusätzlich die programmtechnischen Anweisungen temporär auszubilden, d. h. diese, insbesondere durch Wirkung der Rekonfigurationseinheit, in vorbestimmten Zeitabständen automatisch zu ändern, auch wenn benutzerseitig kein (weiterer) Zugriff auf einen Server erfolgt.

Besonders bevorzugt ist es zudem, das erfindungsgemäße Entstrukturieren durch die Dekonstruktionseinheit so vorzunehmen, dass die entstrukturierten Dokumentkomponenten in der Art einer sog. semantischen Verschlüsselung vorhanden sind mit den Operationen Vertauschen, Entfernen, Hinzufügen und/oder Austauschen von einzelnen, inhaltswirksamen Dokumentkomponenten, z. B. Wörter, Frames, Textseiten usw. Alternativ ist es auch möglich, das erfindungsgemäße Entstrukturieren durch andere Behandlungsformen des elektronischen Dokuments vorzunehmen, beispielsweise das Entfernen von Formatbefehlen oder das Durchführen klassischer Verschlüsselungsoperationen, wie etwa einer XOR-Funktion. Insbesondere, wenn in dem erfindungsgemäßen Entstrukturieren durch die Konstruktionseinheit ein Verschlüsseln liegt, wirkt bevorzugt die programmtechnische Anweisung selbst als Rekonstruktionsanweisung, oder aber beschreibt einen Zugriffspfad für eine serverseitige Datei mit einer solchen Rekonstruktionsanweisung für das verschlüsselte (entstrukturierte) Dokument.

Weiterbildungsgemäß ermöglichen dann die programmtechnischen Anweisungen, die in der beschriebenen Weise vom Server zur lokalen Abarbeitung herangeführt werden, auch ergänzende Funktionalität der benutzerseitigen Zugriffseinheit sowie der zugeordneten Ausgabeeinheit zu steuern bzw. zu realisieren; beispielsweise lässt sich bereits auf diesen Wege die weiterbildungsgemäß vorgesehene Zeitabhängigkeit der zum Benutzer übertragenen programmtechnischen Anweisungen herstellen.

Das erfindungsgemäße Realisieren von (durch Scripte eingefügte) Entschlüsselungsoperationen zur Wiederherstellung eines geschützten Dokuments für einen Benutzer ergibt die zusätzliche Möglichkeit, ergänzend (oder im Rahmen der übertragenen Scripte) Scripte zur Ausgabesteuerung vorzusehen, nämlich insbesondere zur Steuerung bzw. Beeinflussung von clientseitigen Operationen wie Drucken, Abspeichern, Erlauben oder Verhindern von Copy-Paste usw.; innerhalb der Gesamtkomplexität eines bereits mit aufrufbaren Scripten versehenen elektronischen Dokuments dürfte dies nicht weiter auffallen.

Durch eine solche, weiterbildungsgemäß vorgesehene Technologie, nämlich das Miteinander von funktionsbestimmenden Scripten (insbesondere betreffend verschiedene Nutzungs- und Ausgabemodalitäten des Dokuments) mit Entschlüsselungsscripten im Rahmen der vorliegenden Erfindung kann erfolgreich verhindert werden, dass ein unberechtigt Zugreifender (Hacker) einfach durch Deaktivieren einer Ablaufengine für die betreffende Scriptsprache Zugriff nimmt. Da auch zum Zweck der Entschlüsselung bzw. Rekonstruktion notwendig, ist es nicht einfach möglich, die scriptgesteuerte Ausgabekontrolle durch Deaktivieren der Scriptengine zu unterlaufen.

In dieser möglichen Realisierungsform der Erfindung erhalten dann die erfindungsgemäß vorgesehenen programmtechnischen Anweisungen unmittelbaren Einfluss auf die Ablaufumgebung auf Clientseite; neben den ausführlich beschriebenen Scripten bieten sich gerade zu diesem Zweck auch Programm- bzw. Funktionskomponenten, Programmklassen, Methoden oder dergl. zum Ablauf in einer jewei-

ligen Betriebssystemumgebung (z. B. .DLL) als programmtechnische Anweisungen an.

In einer praktischen Realisierungsform der vorliegenden Erfindung wird zudem die das vollständige, brauchbare Dokument anbietende Servereinheit mittels einer vorgeschalteten Proxyservereinheit vom Internet (bzw. einem zugreifenden Benutzer) entkoppelt, dergestalt, dass eine solche Proxyeinheit, etwa mittels ASP (= Activ Server Pages) oder PHP, den erfindungsgemäß dynamisierten Scriptdialog mit dem Benutzer bzw. dessen Zugriffssystem führt, die Benutzersession steuert und auf die erfindungsgemäße Weise den Zugriff auf die Server-Dokumente kontrolliert.

Die vorliegende Erfindung realisiert somit durch die scriptbasierte, dynamisierte Zugriffssteuerung der Zugriffe eines Benutzers auf elektronische Internet-Serverdateien eine wirksame Zugriffskontrolle, die durch Erhöhung der Komplexität des (serverseitigen) Dateizugriffs, der benutzerseitig aufzurufenden Dateien und Dateiverweise sowie der zeitlich begrenzten Gültigkeit von Server-Dateinamen nahezu beliebig verbessert werden kann, auch im Hinblick auf eine durch benutzerseitige Ereignisse bzw. Zugriffshandlungen getriggerte (und damit weiter flexibilisierte) Aktivierung der programmtechnischen Anweisungen zum Ablauf.

Weitere Vorteile, Merkmale und Einzelheiten der Erfindung ergeben sich aus der nachfolgenden Beschreibung bevorzugter Ausführungsbeispiele sowie anhand der Zeichnungen; diese zeigen in

Fig. 1: ein schematisches Blockschaltbild einer Vorrichtung zur geschützten Ausgabe eines elektronischen Dokuments gemäß einer ersten Ausführungsform der vorliegenden Erfindung und

Fig. 2: ein schematisches Blockschaltbild von Funktionskomponenten der Dekonstruktionseinheit der Fig. 1.

Wie in Fig. 1 gezeigt, verbindet ein schematisch durch einen senkrechten Strich symbolisiertes elektronisches Datenübertragungsnetz 10, im vorliegenden Fall das Internet, eine linksgelegene Benutzerseite 12 mit einer rechtsgelegenen Serverseite 14.

Genauer gesagt greift ein Benutzer mittels einer Zugriffseinheit 16 -- üblicherweise ein PC, der mittels geeigneter Hard- und Software zur Internet-Datenkommunikation mit den geeigneten Protokollen vorgesehen ist -- auf eine Servereinheit 18 zu, die im dargestellten Ausführungsbeispiel einen zum Anbieten elektronischer Dokumente, beispielsweise elektronischer Webseiten mit Schrift- und Bildgehalt, eingerichteten Dokumentserver 20 sowie eine diesem netzseitig vorgeschaltete Proxy-Servereinheit 22 aufweist.

Benutzerseitig ist mit der Zugriffseinheit 16 eine Ausgabeeinheit 24 verbunden, die, üblicherweise softwaremäßig, zum Empfangen und Aufbereiten der über das Netz 10 enthaltenen elektronischen Dokumente geeignet ist und hardwaremäßig ein entsprechendes Dokument, etwa über einen Bildschirm oder Druck od.dgl., dem Benutzer anbietet. Gangige, handelsübliche Softwarerealisierungen werden auch als Browser bezeichnet und sind zur Verarbeitung üblicher Dokumentformate für elektronische Dokumente (HTML, XML, diverse Grafik-, Video- und Multimediaformate usw.) geeignet.

Mit den bislang beschriebenen Einheiten 10 bis 24 wird in ansonsten bekannter Weise eine Internet-Datenkommunikation durchgeführt, bei welcher der Benutzer über die Zugriffseinheit durch Eingabe einer entsprechenden Webadresse auf den Dokumentserver zugreift, von diesem dann elektronische Dokumente, etwa als HTML-Dokumente, zurückgesandt bekommt, welche dann mittels der Ausgabeeinheit zur Anzeige auf einem Bildschirm aufbereitet und für den Benutzer betrachtbar sind, bzw. über geeignete Steuermechanismen Auswahl und Abruf weiterer elektronischer Dokumente bzw. Dokumentseiten ermöglichen.

Konkret erfolgt der Aufbau eines elektronischen Dokuments zur benutzerseitigen Betrachtung durch HTML-Befehle eines elektronischen Dokuments, welches der Benutzer nach dem Erstkontakt mit dem Dokumentserver 20 (über die Proxy-Servereinheit 22) empfängt. Durch die Funktionalität der Ausgabeeinheit 24 (bzw. der zugehörigen Browser-Software) wird dann das HTML-Dokument in entsprechenden Text bzw. zugehörige Formate auf einem Bildschirm umgesetzt.

Üblicherweise enthält zudem ein HTML-Dokument auch Pfadangaben bzw. Ladebefehle für weitere elektronische Dateien und Inhalte, wie etwa Grafiken oder Fotos, die nicht unmittelbar mit dem HTML-Dokument geladen werden, sondern die vielmehr bei dem sequentiellen Abarbeiten der einzelnen HTML-Schritte auf Benutzerseite zu einem (oder mehreren) erneuten Aufrufen der Servereinheit führen, mit dem Ziel, auch die neuerlichen, weiteren Dateien auf die Benutzerseite zu laden.

Durch diese wie bislang bekannt ablaufenden Schritte entsteht nunmehr auf der Benutzerseite das elektronische Dokument, zusammengefügt durch eine Vielzahl von Dokumentkomponenten, die entweder aus dem ursprünglichen HTML-Dokument stammen, oder aber die schrittweise durch weitere Serverzugriffe und dadurch erhaltene, weitere Daten dem Dokument hinzugefügt werden. Der der Zugriffseinheit üblicherweise zugeordnete Cache sorgt zudem dafür, dass das so geladene, schon recht komplexe Dokument in ansonsten bekannter Weise gepuffert wird.

Gemäß der vorliegenden Erfindung erfolgt jedoch innerhalb einer vorbestimmten Zeit von beispielsweise einigen Minuten und/oder bei einem erneuten Zugriffsversuch des Benutzers auf die Servereinheit eine Veränderung der konkreten Datei- bzw. Pfadbezeichnungen der Servereinheit, so wie sie mit dem benutzerseitig empfangenen HTML-Dokument, z.B. als Javascript-Anweisungen mit entsprechenden Pfad- und Dateiangaben für den Server, ausgeführt werden. Der erneute Zugriffsversuch mit entsprechend nicht mehr aktuellen Pfad- und Dateiangaben führt zu einem Fehler, so dass

ein erneutes Wiederherstellen des kompletten elektronischen Dokuments auf Benutzerseite nicht möglich ist (und durch die geänderten Pfad- bzw. Dateiangaben hat auch der benutzerseitige Cache keine Möglichkeit, das geladene Dokument aus seinem Pufferspeicher zu rekonstruieren). Vielmehr wäre es notwendig, neue Daten vom Server zu übertragen, die dann zu einem erneuten Aufbau und erneuter Darstellung des Dokuments führen.

Eine besonders geeignete Umgebung ist die bekannte Scriptsprache Javascript, die im gezeigten Ausführungsbeispiel mit ihren Einzelbefehlen bzw. Anweisungen zudem nicht vollständig auf die Zugriffseinheit (zur Abarbeitung durch die Ausgabeeinheit) übertragen wird, sondern die auf der Serverseite verbleibt, und die durch die benutzerseitige Zugriffseinheit lediglich schritt- bzw. befehlsweise abgearbeitet wird.

Hierdurch wird es im Rahmen der vorliegenden Erfindung auch besonders einfach, die jeweiligen, serverseitig zur Verfügung gestellten Javascript-Befehle (etwa als Reaktion auf einen bereits erfolgten Zugriff) zu modifizieren bzw. im Hinblick auf ihren Inhalt zu ändern und so die gewünschte Sicherungswirkung zu erreichen.

Besonders geeignet als vom Server zusätzlich zu übertragender Daten für das benutzerseitig anzuzeigende elektronische Dokument ist zudem das Format XML. Vergleichbar mit HTML, ist XML ein Dokumentformat, welches in der Lage ist, Texte oder weitere Informationen, einschließlich Formatbefehle, zur benutzerseitigen Darstellung zu übertragen, wobei insbesondere XML den Vorteil aufweist, durch Javascript-Befehle besonders einfach und flexibel aufruf- bzw. bearbeitbar zu sein.

Konstruktiv wird die beschriebene Vorgehensweise bei dieser Ausführungsform der Erfindung dadurch realisiert, dass dem Dokumentserver 20 bzw. der Proxyservereinheit 22 eine Dekonstruktionseinheit 26 bzw. eine Rekonfigurationseinheit 28 zugeordnet sind, welche auf eine Speichereinheit 30 zugreifen.

Genauer gesagt wird in der erfindungsgemäßen Weise mittels der Dekonstruktionseinheit 26 ein im Dokumentserver gespeichertes elektronisches Dokument mit seinen einzelnen Dokumentkomponenten so in unzusammenhängende (entstrukturierte) Einzelteile und Einzelkomponenten zerlegt, dass ohne zugehörige Rekonstruktionsanweisungen ein Herstellen eines vollständigen, brauchbaren Dokuments nicht möglich ist. Diese Rekonstruktionsanweisungen, welche insbesondere auch in Form von Javascript-Befehlen, Javascript-Pfadhinweisen, XML-Befehle od.dgl. realisiert sein können, werden in der Speichereinheit 30 abgelegt.

Die der Proxyservereinheit 22 zugeordnete Rekonfigurationseinheit 28 ist nunmehr in der Lage, die erfindungsgemäße Anpassung bzw. Änderung der Befehle vorzunehmen, mit welchen zugriffsseitig die elektronischen Dokumente hergestellt werden bzw. ein Dokumentzugriff erzeugt wird. Mit anderen Worten, die Rekonfigurationseinheit 28 sorgt im vorliegenden Ausführungsbeispiel für die Änderung der Javascript-Befehle so, dass ein erneutes Aufrufen bzw. Zugreifen einen anderen Pfad bzw. einen anderen Dateinamen erzeugt und eine entsprechende Zuordnung bzw. Koordination, entsprechend dem Inhalt der Speichereinheit 30, vornimmt.

Insoweit ist also die Rekonfigurationseinheit 28 als zusätzliche Funktionalitat der Servereinheit zu verstehen, die -- ansonsten bekannte -- statische, d.h. unveränderliche programmtechnische Anweisungen in Form von Javascript-, DHTML- oder anderen Befehlen und Anweisungen in Abhängigkeit von Benutzeraktionen, Zugriffen und/oder Zeitablauf ändert, so dass der erfindungsgemäße Schutzzweck erreicht wird. Im Stand der Technik ist eine solche Einheit nicht vorgesehen, mit der Wirkung, dass dort die entsprechenden programmtechnischen Anweisungen unverändert bleiben. So ist auch das der vorliegenden Erfindung zugrundeliegende Prinzip der "Dynamisierung" der benutzerseitigen, vom Server gesteuerten Anweisungen zu verstehen.

Begreift man das Dekonstruieren bzw. Zerlegen eines elektronischen Dokuments in ohne Rekonstruktion nicht brauchbare Einzelteile als Verschlüsselung, würde nunmehr durch die Javascript-

Befehle die Rekonstruktion und damit die Entschlüsselung möglich werden. Die durch die Rekonfigurierungseinheit 28 jedoch erreichte Dynamisierung der Javascript-Befehle macht wiederum die Möglichkeit zur Entschlüsselung auf Benutzerseite temporär und von einzelnen bzw. einer vorbeschriebenen Anzahl von Zugriffsversuchen abhängig. Damit erhöht sich die Schwierigkeit auf Benutzerseite erheblich, zu einem Zeitpunkt ein vollständiges Dokument, etwa zur unautorisierten Weitergabe oder Speicherung, zu erzeugen, insbesondere wenn manuelle Eingriffe eines Benutzers zum Darstellen eines Dokuments, etwa Mausbetätigung oder Scrollen des Bildschirms, erst die erfindungsgemäße Rekonstruktion ermöglichen.

Genauer gesagt ermöglicht es die vorliegende Erfindung, dass über die von der Benutzerseite zugreifbaren Javascript-Befehle (alternativ können auch DHTML-Befehle od.dgl. abruf- bzw. ausführbar sein) auf Benutzerseite Funktionen in der Form eines Programmes ausgeführt werden, die unmittelbar den Ablauf der Benutzersession beeinflussen bzw. den Inhalt der dem Benutzer zugängigen Ausgabe oder Anzeige beeinflussen. So könnte beispielsweise ein Dateiverweis, welcher von der Zugriffseinheit des Benutzers verfolgt wird, selbst ein Programm beinhalten, welches dann auf der Zugriffseinheit bzw. der Ausgabeeinheit ausgeführt wird und dort abläuft. Damit läßt sich dann die beschriebene Funktionalität, nämlich z.B. der Effekt, dass lediglich auf dem Bildschirm sichtbarer Text unverschlüsselt, der restliche Dokumenttext jedoch verschlüsselt ist, realisieren und der beabsichtigte Schutzzweck weiter erhöhen. Diesem liegt zugrunde, dass üblicherweise derartige, von der Serverseite her mittels des Datei- bzw. Pfadzugriffs herangeführte Programme oder Anweisungssequenzen nicht auf der Benutzerseite gespeichert oder gepuffert werden, so dass eine benutzerseitige Rekonstruktion überaus schwierig, wenn nicht gar unmöglich ist.

Unter Bezug auf die Fig. 2 wird nachfolgend eine weitergehende Realisierungsform der Dekonstruktionseinheit 26 zur Vorbereitung und Verschlüsselung eines Dokuments beschrieben.

Die Fig. 2 zeigt dabei in einer schematischen Blockschaltbild-Darstellung den Aufbau einer Schlüsselerzeugungs- und Verwaltungseinheit mit den zugehörigen Funktionskomponenten im Rahmen der vorliegenden Erfindung, die benutzt werden kann, um durch die Technologie der semantischen Verschlüsselung zu schützende elektronische Dokumente in geschützte Volumendateien etwa HTML-Dateien, und zugehörige Schlüsseldateien (als Grundlage für die Scripte bzw. programmtechnischen Anweisungen) umzusetzen.

Dabei ermöglicht es die im Zusammenhang mit Fig. 2 beschriebene Ausführungsform insbesondere auch, nicht lediglich eine (beim Wiederherstellen zur ursprünglichen, korrekten Datenmenge führende) Schlüsseldatenmenge zu erzeugen, sondern eine Mehrzahl von Schlüsseldatenmengen, so dass auch durch diesen Aspekt des Vorliegens einer Mehrzahl möglicher Schlüssel (von denen auch wiederum einer zu dem auch inhaltlich korrekten, und nicht nur scheinbar korrekten Ergebnis führt) die Sicherheit der vorliegenden Erfindung weiter erhöht werden kann.

Die Fig. 2 soll am Beispiel eines elektronischen Textdokuments beschrieben werden, welches in einem üblichen Format (z.B. Microsoft WORD) vorliegt und mit geeigneten Texteditoren erstellt wurde. Das Textdokument besteht aus dem Satz

Peter geht um 20.00 Uhr zum Bahnhof. Der Zug ist pünktlich.

ist in einer Speichereinheit 52 gemäß Fig. 2 gespeichert und soll in nachfolgend zu beschreibender Weise durch Wirkung der in Fig. 2 gezeigten, weiteren Funktionskomponenten semantisch verschlüsselt werden, um dann im Rahmen der vorliegenden Erfindung dynamisch und Scriptgesteuert wiederherstellbar zu sein.

Eine der Dokumentspeichereinheit 52 nachgeschaltete Lese-/Zugriffseinheit 54, welche mit einer Formatdateneinheit 56 zusammenwirkt, stellt fest, dass das obige, in der Speichereinheit 52 gespeicherte Dokument der Formatstruktur MS-WORD folgt (idealerweise enthält die Formatdateneinheit 56 sämtliche Format- bzw. Strukturinformationen gängiger Datenformate), und greift mit diesen (dateibezogenen) Formatinformationen auf das Textdokument in der Dokumentspeichereinheit 52 zu. Die der Lese-/Zugriffseinheit 54 nachgeschaltete Analyseeinheit 58 ist

nunmehr in der Lage, auf der Basis der von der Leseeinheit 54 gelesenen Dokumentinformationen diese zu analysieren und zu bewerten, wobei die Analyseeinheit 58 zum einen das elektronische Dokument in seine einzelnen Informationskomponenten zerlegt und diese in eine Informationskomponentenspeichereinheit 60 ablegt (im vorliegenden Beispiel wären dies die einzelnen Wörter), und zusätzlich die Dokumentstruktur als Struktur von zwei durch Punkte begrenzten Sätzen erkennt und diese Dokumentstruktur in der Dokumentstrukturspeichereinheit 62 zerlegt ablegt. Insoweit erhält der Inhalt der Einheit 62 den Charakter einer dokumentspezifischen Metadatei, auf die auch spätere Verschlüsselungsvorgänge (auch ggf. nur selektiv) zugreifen können.

Konkret könnte der Inhalt der Dokumentstrukturspeichereinheit nach der Analyse des Ausgangsdokuments durch die Analyseeinheit wie folgt aussehen:

Satz 1 (1, 2, 3, 4) Satz 2 (1, 2, 3),

während die Informationskomponentenspeichereinheit 60 dieser strukturellen Analyse entsprechenden Informationskomponenten, also Worte enthält:

(1.1) Peter
(1.2) geht
(1.3) um 20.00 Uhr
(1.4) zum Bahnhof
(2.1) der Zug
(2.2) ist
(2.3) pünktlich

Mit dieser für das nachfolgende Vornehmen der Verschlüsselungsoperationen wichtigen Vorbereitung ist es nunmehr möglich, sowohl auf die einzelnen Informationskomponenten (im vorliegenden Beispiel die einzelnen Worte), als auch auf die Folgen von Informationskomponenten bzw. Strukturen die Basisoperationen der semantischen Verschlüsselung durchzuführen, nämlich das Vertauschen, Entfernen, Hinzufügen oder Austauschen. Dabei liegt eine wesentliche Schutzwirkung der erfindungsgemäßen semantischen Verschlüsselung darin, dass diese Operationen nicht beliebig durchgeführt werden, sondern dass dies vielmehr unter Beibehal-

tung der grammatikalischen, syntaktischen und/oder formatmäßigen Regeln erfolgt, so dass auch als Ergebnis der Verschlüsselung ein Resultat entsteht, welches scheinbar (d.h. ohne inhaltliche Prüfung) korrekt zu sein scheint, mit anderen Worten, dem man nicht ansieht, dass es sich in der Tat um ein verschlüsseltes Ergebnis handelt.

Im vorliegenden Ausfuhrungsbeispiel wird mit Hilfe der Verschlüsselungseinheit aus dem oben angegebenen elektronischen Dokument der folgende Text:

Thomas kommt um 16.00 Uhr vom Friedhof. Der Zug ist pünktlich.

Ohne Kenntnis des wahren Inhaltes erscheint dieser Satz also wie ein offenes, unverschlüsseltes Ergebnis, so dass eine wesentliche, schutzbegründende Wirkung der vorliegenden Erfindung bereits darin liegt, dass ein Angreifer angesichts dieses Textes möglicherweise gar nicht erst den Eindruck gewinnt, es handle sich um eine Verschlüsselung, und so von Anfang an einen Angriff auf diesen Text unterläßt.

Konkret wurde im vorliegenden Ausführungsbeispiel durch Wirkung einer Äquivalenzeinheit 70 (die in ihrer einfachsten Fassung als Tabelle bzw. Datenbank von äquivalenten, d.h. entsprechenden und austauschbaren, Begriffen verstanden werden kann, folgendes vorgenommen: Die Inhaltskomponente "Peter" des Ausgangsdokuments wurde durch die grammatikalisch äquivalente Inhaltskomponente "Thomas" ersetzt, wobei Satzstruktur und Grammatik beibehalten wurden, der Sinn des Ursprungsdokuments jedoch bereits zerstört ist. Entsprechend wurde die Inhaltskomponenten "geht" des Ursprungsdokuments in die äquivalente Komponente "kommt" ersetzt, die Inhaltskomponente "um 20.00 Uhr" wurde ersetzt durch "um 16.00 Uhr" (hier wurde durch Wirkung der Äquivalenzeinheit festgestellt, dass es sich um ein numerisches Datum in Form einer Uhrzeit handelt, so dass eine Manipulation innerhalb der zulässigen Uhrzeiten möglich war), und die Inhaltskomponente "zum Bahnhof" wurde ersetzt durch die Inhaltskomponente "vom Friedhof". Dabei wurde zudem durch eine ebenfalls mit der Manipulationseinheit 64 verbundene, den geschilderten Verschlüsselungsbetrieb beeinflussende semantische Regeleinheit 72 sichergestellt, dass das Verschlüsselungsergebnis "...kommt ...

vom Friedhof" grammatikalisch und syntaktisch korrekt ist, insoweit also nicht als manipuliert identifiziert werden kann. (Auch das zusätzliche "zum" wäre hier korrekt). Auch wurde mittels der Manipulationseinheit 64 und der zusammenwirkenden Äquivalenzeinheit 70 bzw. semantischen Regeleinheit 72 festgestellt, dass die Inhaltskomponente "der Zug" des nachfolgenden Satzes in einem inhaltlichen Bezug zu der in den vorhergehenden Satz neu eingebrachten Inhaltskomponente "Friedhof" steht, so dass selbst ohne eine Verschlüsselung des zweiten Satzes ein völlig anderer Sinn (und damit ein Verschlüsselungseffekt) entsteht.

Als Ergebnis dieser beschriebenen, einfachen Verschlüsselungsoperationen wird somit das Verschlüsselungsergebnis

"Thomas kommt um 16.00 Uhr vom Friedhof. Der Zug ist pünktlich."

als Volumendaten ausgegeben und in einer Volumendaten-Speichereinheit abgelegt, während ein das Rekonstruieren ermöglichender Schlüssel (im vorliegenden Ausführungsbeispiel eine Information über die jeweils vertauschten Worte mit deren Position im Satz sowie in jeweiligen inhaltlichen Begriffen) in einer Schlüsseldaten-Speichereinheit 74 abgelegt wird. Entsprechend könnte die zugehörige Schlüsseldatei für die Speichereinheit 74 wie folgt aussehen (im folgenden Beispiel wird von der Rekonstruktionseinheit der Scriptbefehl EXCHANGE interpretiert, um die im Argument angegebene Vertauschung durchzuführen):

```
EXCHANGE    (1.1; Thomas)
EXCHANGE    (1.2; kommt)
```

usw.

Hier ist geeignet das Vokabular der Script-Befehlssprache selbst dynamisch, kann etwa durch Funktionen einer Scriptsprache geändert werden; der Befehl EXCHANGE würde so selbst durch einen anderen, beliebigen Ausdruck ersetzt werden können.

Gemaß einer weiteren Ausführungsform der Erfindung ist vorgesehen, eine Mehrzahl von Schlüsseldateien zu erzeugen, von denen jedoch nur eine das korrekte Rekonstruktionsergebnis erzeugt. Schlusseldatei 2 könnte entsprechend wie folgt beginnen:

EXCHANGE (1.1; Rüdiger)
(Rest wie obige Schlüsseldatei);

Schlüsseldatei beginnt mit:

EXCHANGE (1.1; Claus)

usw.

Im Ausführungsbeispiel der Fig. 2 ist zusätzlich diesen beiden Speichereinheiten eine Ausgabeeinheit 78 nachgeschaltet, die in besonders einfacher Weise die Schlüsseldaten 74 in Form eines Scripts aufbereitet und als lauffähige Scriptdatei 84 ausgeben kann; dies geschieht mit Hilfe einer Konvertierungseinheit 80, welche in ansonsten bekannter Weise aus den Volumendaten der Speichereinheit 76 ein der verschlüsselten Fassung entsprechendes (HTML-)Volumendokument 82 erzeugt, und aus den Index- bzw. Rekonstruktionsdaten der Speichereinheit 74 ein selbständig im Rahmen einer geeigneten Ablaufumgebung lauffähige(s) Strukturbeschreibung, Script, z.B. als Javascript, XML, VB-Script od.dgl., und welches dann selbständig beim Ablaufen das als HTML-Datei ausgegebene Volumendokument 82 bearbeitet und in die ursprüngliche, unverschlüsselte und vollständige Form zurückführen kann.

Es versteht sich von selbst, dass dabei die oben lediglich exemplarisch als Ziel der semantischen Operationen behandelten Worte bzw. Sätze auch beliebige andere inhaltsrelevante Komponenten bzw. Inhaltskomponenten eines elektronischen Dokuments sein können, so etwa Bilder, Grafiken, Grafikelemente oder vergrößerte Buchstaben innerhalb einer Seite, Formatbefehle, Tabellen oder andere Strukturelemente. All diese können im Prinzip im Rahmen der vorliegenden Erfindung durch die weiterbildungsgemäß vorgesehenen Operationen der semantischen Verschlüsselung geeignet manipuliert und dann mittels (dynamischer) Scripte auf der Basis der Rekonstruktionsdaten dynamisch wiederhergestellt werden.

Auch wenn sich HTML als besonders geeignetes Format für das Volumendokument 82 eignet (welches dann im Rahmen der Erfindung nach erfolgter, oben exemplarisch beschriebener semantischer Verschlüsselung dem erfindungsgemäßen endstrukturierten elektronischen Dokument entspricht), so ist für ein entsprechendes Dokumentformat prinzipiell jegliches Format denkbar, welches zusammen mit den erfindungsgemäßen programmtechnischen Anweisungen (Scripten) auf der benutzerseitigen Zugriffseinheit empfang- und darstellbar ist.

Zusätzlich ist die in Fig. 2 schematisch gezeigte Ausführungsform geeignet, nicht nur eine Schlüsseldatei für die Speichereinheit 74 (bzw. als lauffähige Scriptdatei 84) zu erzeugen, sondern eine Mehrzahl von diesen, von denen idealerweise jedoch wiederum nur eine zu einem inhaltlich tatsächlich korrekten Ergebnis führt, während andere Schlüsseldateien als Scripte einen Entschlüsselungsvorgang auslösen, welcher zwar ebenfalls zu einem sinnvollen (und damit scheinbar korrekten) Ergebnis führt, inhaltlich jedoch nicht mit der Ursprungsfassung übereinstimmt. Hierdurch ist dann eine weitere Erhöhung der Verschlüsselungssicherheit gegeben. Dabei sollte es unmittelbar einsichtig sein, dass bereits geringe inhaltliche Abweichungen den (für einen Nutzer eigentlich wertbildenden) Sinn des Ursprungsdokuments vollständig zerstören, so dass es möglicherweise nur geringer Modifikationen bzw. einer geringen Anzahl von Verschlüsselungsoperationen (mit der Folge einer entsprechend kurzen Scriptdatei als Schlüsseldaten) bedarf, um den vorgesehenen Schutzzweck zu erreichen, bis hin zur bereits erwähnten Nicht-Verschlüsselung der Ursprungsdatei, die ihren Schutzzweck lediglich aus dem Umstand herleitet, dass ein unberechtigt Zugreifender die Unsicherheit hat, ob er es mit einem offenen (d.h. der Ursprungsdatei auch entsprechenden) Inhalt, oder aber mit einem verschlüsselten, d.h. nicht mit der Ursprungsdatei übereinstimmenden Inhalt zu tun hat.

Ein besonders eleganter Weg, eine Mehrzahl von (ähnlichen) Schlüsseldateien zu erzeugen, besteht darin, eine scriptartige Schlüsseldatei zu verwenden, die mit zugeführten Parametern

(z.B. Index- bzw. Reihenfolgeangaben) verschiedene Entschlüsselungsergebnisse liefert, wobei im Rahmen der beschriebenen Weiterbildung der Erfindung alle Ergebnisse scheinbar korrekt sind, jedoch nur eines inhaltlich vollständig dem Original entspricht. Indem dann z.B. diese zugeführten Parameter im Wege einer ansonsten bekannten zyklischen Permutation eine Reihenfolgemanipulation auf ganze Sätze (bei einem Textdokument) ausführen, wäre genau dieser Zweck erreicht: Eine Mehrzahl von Schlüsseldateien erzeugt ein scheinbar korrektes Ergebnis, jedoch nur eine Datei enthält die wirklich korrekte Reihenfolge der erfindungsgemäß und scriptgesteuert ver- bzw. entschlüsselten Sätze als Textkomponenten.

Die vorliegende Erfindung ist, wie erwähnt, nicht auf das exemplarische Beispiel von Textdateien beschränkt. So bietet es sich insbesondere auch an, jegliche weiteren elektronischen Dokumente durch die prinzipiell beschriebene Weise zu verschlüsseln, solange diese elektronischen Dokumente eine für die Basisoperationen des Vertauschens, Entfernens, Hinzufügens oder Austauschens geeignete Struktur aus Inhaltskomponenten aufweisen. Typische Anwendungsfälle sind insbesondere Musikdateien, die üblicherweise im MP3-Format vorliegen, und wo es im Rahmen der vorliegenden Erfindung möglich ist, die durch das MP3-Format vorgegebenen Datenstrukturen (sog. Frames) einzeln oder blockweise (idealerweise auch takt- oder abschnittsweise, bezogen auf das jeweilige Musikstück) auszutauschen, zu entfernen oder zu vertauschen. Entsprechendes gilt für Bild- und/oder Videodateien, denn auch die dort gängigen, bekannten Dokumentformate basieren auf einer Folge von Frames als Inhaltskomponenten (bei Bildern oder elektronischen Videos sind dies die jeweiligen Einzelbilder bzw. inter- oder extrapolierende Frames), die in der erfindungsgemäßen Weise manipuliert werden können.

Weitere mögliche und günstige Weiterbildungen der Erfindung sehen vor, dass eine Rekonstruktionsdatei in Form eines Scripts in einem ASCII- und/oder HTML-Dateiformat vorliegt. Insbesondere im Hinblick auf eine Client- und/oder Servereinheit schützende

Firewall bieten sich damit vereinfachte Möglichkeiten, eine solche Firewall unbehelligt zu durchdringen.

Eine weitere, vorteilhafte Weiterbildung der Erfindung sieht vor, eine Script-Rekonstruktionsdatei geeignet in elektronische Dokumentdaten (desselben oder eines anderen Dateityps) einzubetten, und zwar so, dass damit Format und (wiedergegebener) Inhalt einer solchen Gastdatei unverändert bleibt; in besonders vorteilhafter Weise bietet sich für eine solche, verdeckte Weitergabe von Rekonstruktionsdateien, mit dem Zweck einer weiteren Sicherheitserhöhung, daher ein Bereich der Gastdatei an, welcher nicht unmittelbar inhaltswirksam ist, also z.B. Kommentar- oder Informationsbereiche usw.

Insbesondere die Realisisierungsmöglichkeit der erfindungsgemäßen Rekonstruktionsdateien als Scripte bieten zahlreiche Vorteile: So ermoglicht das scriptgesteuerte Zusammenführen im Rahmen der vorliegenden Erfindung das Flexibilisieren bzw. weitere Erhöhen der Sicherheit dadurch, dass nicht allein eine Scriptdatei als Rekonstruktionsdatei das (dynamische) Wiederherstellen der unverschlüsselten Form des elektronischen Dokuments durch das Zusammenführen ermöglicht, sondern eine Mehrzahl von Scripten als Rekonstruktionsdateien notwendig ist, welche z.B. vorbestimmte zeitliche Abschnitte des elektronischen Dokuments abdecken und sich dann sukzessive aufrufen. Als Beispiel könnte hier die Erfindung so realisiert werden, dass jeweils eine Scriptdatei als Rekonstruktionsdatei für einen Zeitabschnitt von etwa 30 Sekunden eines MP3-Musikstücks das Rekonstruieren ermöglicht, und dann eine weitere Rekonstruktion das (wiederum scriptgesteuerte) Aufrufen einer nachfolgenden, weiteren Scriptdatei zur Rekonstruktion notwendig macht. Neben einer gesteigerten Sicherheitswirkung ergeben sich hiermit Möglichkeiten einer kontextabhängigen Generierung bzw. Rekonstruktion des ursprünglichen Originaldokuments, einschließlich gar der Möglichkeit, verschiedene Varianten des Originaldokuments kontextabhängig und gezielt wieder herzustellen.

Wesentliche Aufgabe der mit der in Fig. 2 dargestellten Dekonstruktionseinheit 26 zusammenwirkenden Rekonfigurationseinheit 28 ist es nunmehr, die als Ergebnis der vorbeschriebenen Verschlüsselungsoperation erzeugten Scripte (Datei 84 in Fig. 2) zu dynamisieren, d. h. mit Hilfe dieser Scripte ein stets variables Wiederherstellen des Originaldokuments auf Benutzerseite, herbeizuführen.

Besonders geeignet kann dies dadurch geschehen, dass, im Zusammenwirken mit der Konvertierungseinheit 80 (Fig. 2) der Dekonstruktionseinheit 26 kontinuierlich verschiedene Scriptdateien 84 erzeugt und mit den zugrunde liegenden Rekonstruktionsdaten so verknüpft werden, dass nur (noch) eine jeweils aktuelle Fassung einer Scriptdatei 84 clientseitig (d. h. durch Ablauf bzw. Ausführung in der Zugriffseinheit 16, zu dem korrekten elektronischen Dokument in der Wiederherstellung führt, ggf. ist auch auf Serverseite hierzu eine weitergehende Interaktion zwischen Rekonfigurationseinheit 28 und Dekonstruktionseinheit 26 notwendig, nämlich dahingehend, dass auch regelmäßig Volumen- und Rekonstruktionsdaten geändert bzw. angepasst werden müssen.

Da ein Anwendungszweck der vorliegenden Erfindung darin liegt, über den erfindungsgemäßen Schutz- bzw. Sicherungseffekt eine Kontrolle des serverseitigen Anbieters auf Benutzerzugriffe zu erreichen und dahinter insbesondere auch eine kommerzielle Verwertungsabsicht eines Anbieters stehen kann, nur autorisierten Benutzern bzw. nach einem vorhergehenden Transaktionsvorgang bestimmte Dateiinhalte und/oder Operationen zugänglich zu machen, ist serverseitig eine Identifikations- und Abrechnungseinheit 32 vorgesehen, welche in ansonsten bekannter Weise mit einer entsprechend benutzerseitig der Zugriffseinheit 16 zugeordneten Identifikations-, Authentifizierungs- und Abrechnungseinheit 32 zusammenwirkt. Genauer gesagt kann nach einem entsprechenden Übermitteln geeigneter Zahlungsinformationen eines Benutzers, beispielsweise von Kreditkarten-Daten, die Möglichkeit geschaffen werden, dass der Benutzer besondere Zugriffs- oder Nutzungsrechte an einem elektronischen Dokument erhält, welche ansonsten für einen unautorisierten Benutzer in der vorgeschriebenen Art unmöglich gemacht werden. Entsprechendes gilt

auch für eine serverseitig in der Servereinheit vorsehbare, in der Figur nicht gezeigte Benutzergruppen- und/oder Rechteverwaltung, die ein entsprechendes Management von Zugriffs- und Nutzungsrechten vornehmen kann und etwa nur Zugehörigen zu einer Benutzergruppe Zugriff auf bestimmte Dokumente in entschlüsselter Form ermöglicht.

Für eine Authentifizierung bzw. Identifikation eines Benutzers kann es dabei insbesondere günstig sein, Zugriffsdaten bzw. Scriptinformationen, mit welchen der Benutzer selbst auf die Serverseite zugreift, dahingehend auszuwerten, ob diese -- authorisiert erworben -- aus (ggf. unmittelbar) vorhergehenden, ggf. zeitnahen Sessions stammen und damit einen ordnungsgemäßen Zugriff kennzeichnen.

Die vorliegende Erfindung ist nicht darauf beschränkt, die erfindungsgemäßen programmtechnischen Anweisungen über Browser od.dgl. Internet-Zugangssysteme ablaufen zu lassen. Vielmehr ist es insbesondere auch von der vorliegenden Erfindung umfasst, weitergehende, dokumentspezifische Umgebungen durch die erfindungsgemäß modifizierten (rekonfigurierten) programmtechnischen Anweisungen zu steuern bzw. inhaltsmäßig zu beeinflussen, die nicht spezielle Internet-Browser sind. So ist es beispielsweise von der Erfindung mitumfasst, Textverarbeitungsprogramme zur Realisierung der Erfindung mit einer speziellen Ablaufeinheit zu versehen, die dann die erfindungsgemäße Funktionalität zur Rekonstruktion der erfindungsgemäß verschlüsselten Daten und elektronischen Dateien herbeiführt (ein konkretes Beispiel könnte in einer Textverarbeitung bestehen, welche über eine Ablaufeinheit zum Ausführen einer den Verschlüsselungseffekt bewirkenden Script- bzw. Makrosprache verfügt, etwa VBasic).

Als potentielle Weiterbildung eines derartigen, allgemeinen Ansatzes bei der Verwendung einer (generischen) Dokumentdarstellungseinheit zusammen mit einer Ablaufeinheit für die Scripte bietet es sich besonders an, die Ausführung bzw. den Start der Scripte nicht automatisch, wie es etwa im Fall eines Internet-Browsers der Fall ist, an das Laden bzw. Aufrufen eines betref-

fenden Scripts durch eine HTML-Seite zu koppeln, sondern vielmehr den Scriptaufruf von Ereignissen (sog. Events) in der jeweiligen Ablaufumgebung bzw. Ablaufsoftware auf Clientseite abhangig zu machen. So bietet es sich insbesondere an, gewisse, das Aussehen, die Struktur-, Text- oder Bildzusammenstellung eines elektronischen Dokuments auf Clientseite durch ein Script dadurch zu verandern, dass das diese Manipulation herbeiführende Script erst als Reaktion auf ein solches Ereignis in der Ablaufumgebung der Clientsoftware gestartet wird; typische Ereignisse sind etwa die Ereignisse "onhide" oder "onshow" innerhalb des Objektmodells von Microsoft-Windows bzw. des Internet-Explorers, vergleichbar einem Interrupt: Erst als Reaktion auf das Zeigen (bzw. Verstecken) bestimmter Dokumentpassagen würden die mit diesen Events verknüpften Scripte angestoßen und so zur zusätzlichen Sicherungswirkung im Rahmen der vorliegenden Erfindung beitragen. Diesem weiterbildenden Erfindungsgedanken liegt dabei der Ansatz zugrunde, dass insbesondere bei einem Scriptaufruf durch ein HTML-Dokument beim Laden ein quasi-statischer Zustand auftritt, sobald die Scripte ausgeführt worden sind, und durch (missbräuchlichen) Zugriff auf das DOM (Document Object Module) des jeweiligen Dokuments könnte so theoretisch ein Angriff erfolgen; durch eine event-gesteuerte Dynamisierung, insbesondere bei Darstellungs- bzw. Anzeigeoperationen auf Clientseite nach dem vollständigen Laden des Dokuments vom Server, kann dagegen die DOM-Umgebung dynamisiert und damit effektiver vor missbräuchlichem Zugriff geschützt werden.

Eine Weiterbildung könnte dieser Gedanke dadurch erfahren, dass das elektronische (Gesamt-) Dokument als Zusammenstellung verschiedener Hyperlinks verstanden wird, nämlich realisiert als Menge von (wiederum scriptgesteuert aktivier- bzw. veränderbaren) Verweisen auf andere Dokumentbereiche bzw. Inhaltselemente. Entsprechend führt eine Aktivierung des Hyperlink (Event "ondick") zu einer Änderung des Dokuments im DOM.

Ein auf diesem Weg erreichter Schutz vor automatisiertem Auslesen aus dem DOM eines elektronischen Dokuments ist dann besonders schwierig, wenn die gesamte Ausgabe (z.B. auf einem Bild-

schirm) sofort sichtbar gemacht werden kann. In einem solchen Fall kann es keine Bildschirmbereiche geben, die bei Operationen wie "onhide" oder "onshow" ein Event liefern, welches dann in der erfindungsgemäßen Weise eine dynamische Änderung der Werte innerhalb des "DOM" bewirken könnte.

Um auch in einem solchen Fall eine durch Events getriggerte Aktivierung von Scripts erreichen zu können, werden Werte in das DOM an Positionen hinzugefügt, die typischerweise außerhalb eines sichtbaren Bereichs (etwa auf einer Bildschirmdarstellung) erscheinen würden, jedoch niemals wirklich sichtbar gemacht werden konnen. Falls dann jedoch, etwa durch Betätigen eines Scrollbar durch den Benutzer, das Event "onshow" eines derartigen, hinzugefügten Objektes erzeugt wird, erfolgt die Aktivierung eines Scripts und mithin eine Änderung der Werte im DOM so, dass z.B. ähnliche Objekte an einer anderen Stelle außerhalb des sichtbaren Bereichs und innerhalb des DOM eingefügt würden. Vorteilhaft könnten auch diese Objekte wiederum Scripte enthalten, die nach Ausschüttung eines darin enthaltenen Events "onshow" das betreffende Objekt zum Verschwinden oder Verschieben bringen.

Ein analoger Ansatz im Rahmen einer Weiterbildung der Erfindung kann auch auf eine Unter- bzw. Submenge des DOM angewendet werden: Ein Dokument könnte hier als Menge von Flächenelementen angesehen werden, die zusammen das Gesamtdokument darstellen, wobei jedoch innerhalb der einzelnen Flächenelemente Bereiche liegen, die nicht sichtbar sind, weil sie sich außerhalb der sichtbaren Fläche positionieren würden. Falls etwa mit Hilfe einer Operation "mouseOver" derartige Flächen aktiviert würden, kann ein zugehöriger Event die Daten in der entsprechenden Submenge des DOM geändert haben.

Für den Sicherheitsaspekt der vorliegenden Erfindung bedeutet dies, dass ein unberechtigt Zugreifender (Hacker) diese Art von Informationen nur noch kontextabhängig verwenden, wobei ihm jedoch erst nach einer ausfuhrlichen Analyse der Kontext deutlich werden kann.

Ein sich aus der vorliegenden Erfindung ergebener praktischer Vorteil liegt zudem darin, dass, vgl. die Konfiguration und das Zusammenwirken der Funktionseinheiten gemäß Fig. 1, Dekonstruktionseinheit 26 sowie Rekonfigurationseinheit 28, die Einheiten dynamisch miteinander arbeiten und insoweit die Speichereinheit 30 lediglich als Puffer zu verstehen ist. Sobald nämlich, z.B. in Folge Zeitablaufs und/oder Aktion eines Benutzers, etwa ein erneuter Zugriffsversuch, eine erneute (andere) Verschlüsselung eines zugehörigen elektronischen Dokuments erfolgt, werden zwar zugehörige Rekonstruktionsdaten in Form entsprechender programmtechnischer Anweisungen für bzw. durch die Rekonfigurationseinheit 28 erzeugt, eine weitergehende Speicherung, etwa auf Benutzerseite, erfolgt jedoch nicht. Damit wird dann nicht nur Speicherplatz gespart, sondern insbesondere auch eine Aktualisierung eines entsprechenden elektronischen Dokuments erleichtert.

Im weiteren soll noch eine weitere, konkrete programmtechnische Realisierung einer Ausführungsform der vorliegenden Erfindung beschrieben werden, wie sie etwa unter Zuhilfenahme von XML und Javascript realisiert sein könnte (es werden nur die wesentlichen Befehle bzw. deren Abläufe dargestellt).

Es wird dabei davon ausgegangen, dass der Benutzer von der Servereinheit als Startdokument eine HTML-Dokumentseite lädt, welche einen Aufruf für ein Javascript (".js" als Dateikennzeichnung) enthält und selbige Javascript-Programmanweisungen wiederum auf dem Server stehen und von diesem abrufbar sind.

Ein entsprechender, ausschnittsweiser Programmcode einer entsprechenden Javascript-Programmsequenz, die dann benutzerseitig von der Zugriffseinheit schrittweise abgearbeitet wird, könnte etwa wie folgt aussehen:

```
if session ("countjs") = "0" then
   session ("countjs") = "1"
XMLDoc.async = false;
```

```
// berechnen des dynamischen Dokumentnamens schedule.xml2.asp
XMLDoc.load ("schedule.xml" + (1 + 1) + ".asp");
        XSLDoc.async = false;
        XSLDoc.load ("schedule.xsl.asp");
// parsen der XML-Datei
// hier ist die geladene Datei das neue Element
   result = XMLDoc.documentElement.transformNode
(XSLDoc.documentElement);
//Ersetzen des verschlüsselten Textes
  verschlüsselt.innerHTML = result;
else
  alert("*** Fehlermeldung ***");
end if.
```

Erkennbar wird hier eine Überprüfung vorgenommen, ob die vorliegende Anweisungssequenz (Script) bereits geladen worden ist, und wenn dies der Fall ist, erfolgt eine entsprechende Fehlermeldung.

Durch den Befehl XMLDoc.load erfolgt dann das Laden des dynamisch berechneten Dokumentnamens, im vorliegenden Fall des Dokumentnamens schedule.xml2.asp, und mit der result-Anweisung wird dann der Inhalt der Datei schedule.xml2.asp in das vorliegende Dokument eingefügt.

Im vorliegenden Beispiel enthält das zusätzlich aufgerufene Modul schedule.xml2.asp lediglich Textbausteine, die dann geeignet aufgerufen werden; wie bei dem oben gezeigten Script kann jedoch auch hier mittels etwa einer if-Anweisung ein mehrfaches Lesen derselben Datei wiederum unterdrückt werden.

Im Ergebnis wird also durch das gezeigte Beispiel folgendes erreicht: Zum einen sorgt das oben dargestellte Script dafür, dass es nur einmal ausgeführt werden kann; bei einem weiteren Zugriffsversuch würde dieser scheitern und statt dessen zu einer Fehiermeldung führen. Zum zweiten wird mittels des gezeigten Scripts und einer einfachen Rechenoperation (1 + 1) ein weiterer Dateiname dynamisch erzeugt, nämlich der Dateiname

schedule.xml2.asp, auf welchen dann zugegriffen wird, und welcher dann den gewünschten Text zum Einbau in das Dokument liefert. Auch dieser dynamisch erzeugte Dateiname selbst ist jedoch dynamisch und temporär, so läßt sich diese Datei ebenfalls nur einmal aufrufen, und, wie für den Fachmann offensichtlich ist, bietet die gezeigte Vorgehensweise verschiedenste Möglichkeiten, Erzeugung und/oder Berechnung eines solchen Dateinamens zu variieren.

Eine weitere mögliche Weiterbildung der Erfindung liegt darin, dass mit Hilfe von Verschlüsselungsverfahren eine Mehrzahl verschiedener Verweis- bzw. Hyperlinknamen erzeugt wird, die jedoch serverseitig nach einer Entschlüsselung jeweils so interpretiert werden können, dass sie zum gleichen Ziel führen. So wird beispielsweise zu diesem Zweck der Hyperlink auf eine (serverseitig vorhandene) Datei durch eine definierte Kennung, bevorzugt zufallsgesteuert, erweitert, etwa dadurch, dass der konkrete Dateiname durch eine vorbestimmte Anzahl von zufällig bestimmten Zeichen ergänzt wird (und damit statt z. B. 10 Zeichen eine Zeichenfolge von 17 Zeichen aufweist). Nach einer Verschlüsselung entsteht hieraus ein eigenständiger, vollständig neuer verschlüsselter Pfadname, der auch als solcher clientseitig übermittelt werden kann bzw. vom Client zum Aufruf einer nächsten Dokumentseite oder Dokumentkomponente benutzt werden kann.

Serverseitig könnte dann dieser verschlüsselte Pfadname wiederum entschlüsselt werden, und durch Entfernen der letzten sieben Stellen entsteht wiederum der ursprüngliche, zu dem Ziel führende einzige Name. Diese im Rahmen einer Weiterbildung der Erfindung vorgesehene Maßnahme hätte den Vorteil, dass das Erzeugen einer Vielzahl von möglichen Pfadangaben zur Erhöhung der Sicherheit gegen missbräuchliche Zugriffe von Clientseite vereinfacht ist, und bevorzugt wäre zudem, etwa durch ein Fortzählen angehängter Zeichenstrings an den Originalpfad, auch noch das Erkennen nicht mehr gültiger Pfadangaben bzw. Dateianforderungen durch den Client -- für diesen jedoch völlig intransparent -- von der Serverseite her möglich. Je nach Situation kann

ein symmetrischer oder asymmetrischer Schlüssel verwendet werden.

Gemäß einer weiteren bevorzugten Ausführungsform werden im Rahmen der erfindungsgemäßen programmtechnischen Anweisungen (Scripte) Algorithmen eingesetzt, die verschiedenen Konfigurations- und möglichen Angriffssituationen von einer Clientseite her gerecht werden können. So ist es zunächst durch eine serverinduzierte Scriptabfrage möglich, festzustellen, ob, und wenn ja, welche Scriptsprache clientseitig überhaupt verstanden wird, bzw. verwendet werden kann. Zusätzlich kann etwa eine bestimmte Betriebssystemumgebung bzw. Plattform auch auf Clientseite getestet werden. Weiterhin ist es möglich, durch ein Script (welches clientseitig bereits empfangen wurde und zu verschiedenen, bevorzugt variablen Zeitpunkten gestartet wird) festzustellen, ob ein betreffender Client mit dem entsprechenden Server überhaupt noch in einem Online-Kontakt steht, oder aber ob eine Offline-Situation (welche potentiell für Angriffe gefährlicher ist) vorliegt; entsprechend dieser Feststellung könnten sich durch die Scripte ausgelöste Effekte (z. B. Modifikationen in der Darstellung) ändern. Ein besonders einfacher Weg zur Feststellung, ob ein Online-Kontakt aktuell vorliegt, könnte scriptgestalt dadurch implementiert werden, dass innerhalb des Scriptes eine Anfrage an den Server gerichtet wird und dann eine konkrete Antwort erwartet wird (sogenanntes Challenge und Response).

Grundsätzlich liegt es im Rahmen der vorliegenden Erfindung, durch das Vorsehen der programmtechnischen Anweisungen nicht nur ein korrektes (Wieder-)Herstellen des elektronischen Dokumentes zu ermöglichen, auch ist es -- scriptgesteuert -- möglich, inhaltliche und formatmäßige Modifikationen an dem Dokument vorzunehmen, insbesondere auch an einer jeweils rekonstruierten Dokumentseite. So ist es insbesondere möglich, kontinuierlich verschiedene Pfadangaben bzw. Hyperlinks in einen HTML-Code zu integrieren und damit das erneute Laden der Datei (durch einen automatischen Lesevorgang) zu erzwingen; es bestehen somit keine automatisch benutzbaren Hinweise auf eine nächste oder vorige Seite. Eine leichte Variation einer rekonstruierten Seite gegen-

über einer vorhergehenden Rekonstruktion (z. B. eine Frontänderung oder dergl.) verhindert darüber hinaus, dass ein zum Zweck eines missbräuchlichen Zugriffs bzw. einer Entschlüsselung ausgeübter Algorithmus eine konkrete Beziehung zwischen Seiten herstellen kann, ggf. zwei inhaltlich gleiche Seiten als gleich (bzw. identisch) zu identifizieren.

Darüber hinaus können uneinheitliche Strukturelemente bzw. ein uneinheitliches Layout eines rekonstruierten (bzw. scheinbar rekonstruierten) Dokuments durch entsprechende Scriptsteuerung ein Hinweis an einen Benutzer (eines illegal erworbenen Dokuments) sein, dass ein jeweiliger Dokumentinhalt durch mögliche sinnverändernde Manipulationen verschlüsselt wurde.

Zur weiteren Erhöhung der (Sicherheit bietenden) Komplexität im Rahmen der Erfindung können zusätzliche Inhaltselemente als sog. Fallen mittels Scriptcode bzw. scriptgesteuert in das elektronische Dokument integriert werden, wobei diese Inhalte nur durch inhaltliche Prüfung durch einen Benutzer und nicht maschinengesteuert erkannt werden können (da sie z.B. aus einer manuell hergestellten Datei zufällig dem Scriptcode hinzugefügt wurden). Bei der ordnungsgemäßen Darstellung des elektronischen Dokuments findet -- scriptgesteuert -- ein Darstellen dieser zusätzlichen Inhaltskomponenten (Fallen) nicht statt, so dass die ordnungsgemäße Benutzung insoweit für einen Benutzer störungsfrei ist. Dagegen müßte ein unberechtigt Zugreifender bzw. ein Hacker, der das elektronische Dokument in unberechtigter Weise weitergeben will, das Dokument bzw. den Scriptcode inhaltlich prüfen und dann die zusätzlichen Komponenten manuell entfernen.

Generell ist es von der vorliegenden Erfindung umfasst, eine größtmögliche Flexibilität beim Umgang mit dem Laden und Starten von Scripten bzw. scriptgesteuerter Daten auf Clientseite vorzusehen; insbesondere sind von den erfindungsgemäß vorgesehenen programmtechnischen Anweisungen auch Manipulationen bzw. Beeinflussungen einer clientseitigen Programmfunktionalität der Darstellungs- bzw. Ablaufeinheit umfasst, so etwa ein mögliches Aktivieren bzw. Deaktivieren einer Druck-, Kopierfunktion oder

dergl., oder aber eine gezielte Steuerung und Manipulation verschiedener Darstellungsebenen (Layer) auf einer geeigneten Bildausgabeeinheit.

Damit dürfte dann die vorliegende Erfindung sogar den generellen Gedanken umfassen, dass -- scriptgesteuert -- die Funktionalität der benutzerseitigen (clientseitigen) Zugriffseinheit bzw. der zugeordneten Ausgabeeinheit unmittelbar programmtechnisch beeinflussbar ist, wobei, neben Scripten, hier als programmtechnische Anweisungen insbesondere auch (ausführbare) Programmmodule, Programmklassen oder dergl. übertragen und clientseitig gestartet werden können.

Gerade im Hinblick auf einen Offline-Betrieb ergibt sich aus diesem Gedanken, dass ein Offline darzustellendes Dokument direkt in einer mit serverseitig vorgegebenen Scripten geschützten Offline-Darstellungskomponente dargestellt werden kann, oder aber eine solche offline-Darstellungskomponente, in der Art einer Offline (d.h. lokal) vorhandenen Servereinheit, enthält selbst eine Scripterzeugungseinheit, die in der Lage ist, im Rahmen der vorliegenden Erfindung Scripte zu erzeugen, um Dokumente geschützt und scriptabhängig darzustellen bzw. einen reproduzierbaren Angriff über einen Scriptdebugger bzw. über das DOM zu verhindern.

PATENTANSPRÜCHE

1. Vorrichtung zur geschützten Ausgabe eines elektronischen Dokuments über ein bevorzugt öffentliches Datenübertragungsnetz, insbesondere das Internet, mit:
   - einer benutzerseitigen Zugriffseinheit (16) für ein Zugreifen über das elektronische Datenübertragungsnetz (10) auf eine das elektronische Dokument zum Ladezugriff anbietende Servereinheit (18) und
   - einer der benutzerseitigen Zugriffseinheit zugeordneten Ausgabeeinheit (24) zum Empfangen und Ausgeben des elektronischen Dokuments in einer für einen Benutzer vorgesehenen Form,
   - wobei das empfangene Dokument mindestens eine programmtechnische Anweisung aufweist, die für das Ausgeben durch die Ausgabeeinheit ausführbar ist und eine Bezeichnung einer Datei und/oder eines Pfades im Datenübertragungsnetz und/oder einer Darstellungsform des elektronischen Dokuments enthält, wobei mittels der Bezeichnung Dokumentkomponenten des elektronischen Dokuments geladen und/oder geändert werden,

   dadurch gekennzeichnet, dass
   - das elektronische Dokument mittels einer serverseitigen Dekonstruktionseinheit (26) so vorbereitet und hinsichtlich der Dokumentkomponenten entstrukturiert ist, dass es in der für den Benutzer vorgesehenen Form erst nach dem Ausführen einer zugeordneten, durch die Dekonstruktionseinheit entsprechend erzeugten programmtechnischen Anweisung brauchbar ist,
   - und eine der Servereinheit zugeordnete Rekonfigurations-Einheit (28) so ausgebildet ist, dass die programmtechnische Anweisung und/oder die Dokumentkomponenten des empfangenen Dokuments so gebildet oder verändert werden können, dass ein erneutes Empfangen des elektronischen Dokuments durch den Benutzer nach einer vorbestimmten, begrenzten Anzahl von weiteren Zugriffen, insbesondere nach nur einem weiteren Zugriff, eine

Änderung der Anweisung hinsichtlich einer zugehörigen Dateibezeichnung und/oder einer Pfadbezeichnung und/oder einer Darstellungsformbezeichnung, und/oder eine Änderung der damit zu ladenden Dokumentkomponenten, gegenüber einem vorhergehenden Empfangen und Ausgeben, bewirkt.

2. Vorrichtung nach Anspruch 1, dadurch gekennzeichnet, dass die programmtechnische Anweisung ein Element einer zu einer elektronischen Verarbeitung geeigneten Scriptsprache ist, die durch die Ausgabeeinheit bevorzugt schrittweise bearbeitet und/oder ausgeführt werden kann.

3. Vorrichtung nach Anspruch 1 oder 2, dadurch gekennzeichnet, dass die programmtechnische Anweisung mittels Javascript und/oder Visual Basic Script und/oder XML und/oder XSL und/oder HTML realisiert ist, wobei bevorzugt eine Mehrzahl von Anweisungen serverseitig zum Aufrufen und Ausführen durch die benutzerseitige Ausgabeeinheit bereitgestellt wird.

4. Vorrichtung nach einem der Ansprüche 1 bis 3, dadurch gekennzeichnet, dass die programmtechnische Anweisung zum benutzerseitigen Zugreifen auf eine serverseitige Datei ausgebildet ist, deren Dateninhalt mit der programmtechnischen Anweisung zum Zusammenfügen des elektronischen Dokuments in der für den Benutzer vorgesehenen Form vorgesehen ist, wobei bevorzugt die serverseitige Datei eine XML-Datei und/oder eine XML-Datenformatdatei ist.

5. Vorrichtung nach einem der Ansprüche 1 bis 4, dadurch gekennzeichnet, dass die programmtechnische Anweisung so ausgebildet ist, dass vorbestimmte Betriebszustände der benutzerseitigen Zugriffseinheit, insbesondere Betriebssystemparameter, mögliche ablauffähige Scriptsprachen, On- oder Offline-Status, erfasst und von dieser Erfassung abhängig vorbestimmte Befehle der programmtechnischen Anweisung zum Ablauf gestartet werden können, wobei bevorzugt ein Wieder-

herstellen des elektronischen Dokuments in eine brauchbare Fassung kontext- und/oder zeitabhängig erfolgt.

6. Vorrichtung nach einem der Ansprüche 1 bis 5, dadurch gekennzeichnet, dass die Rekonfigurationseinheit zum Zusammenwirken mit einer der Ausgabeeinheit zugeordneten, manuell durch einen Benutzer betätigbaren Bedieneinheit vorgesehen und so ausgebildet ist, dass als Reaktion auf eine vorbestimmte Betätigung des Benutzers, insbesondere eine Änderung einer Maus- oder Zeigerposition auf einem Bildschirm oder eine Änderung eines Bildausschnittes, die programmtechnische Anweisung neu gebildet oder verändert wird, oder die programmtechnische Anweisung zum Erzeugen des elektronischen Dokuments in der für den Benutzer brauchbaren Form ausgeführt wird.

7. Vorrichtung nach einem der Ansprüche 1 bis 6, gekennzeichnet durch eine serverseitige Identifikations- und Abrechnungseinheit (32), die zum Zusammenwirken mit einer der benutzerseitigen Zugriffseinheit (16) zugeordneten Identifikations- und Abrechnungseinheit vorgesehen und so ausgebildet ist, dass als Reaktion auf einen erfolgten Identifikations-, Authentifikations- und/oder Transaktionsvorgang mit einem Benutzer die Rekonstruktionseinheit die programmtechnische Anweisung und/oder die Dokumentkomponenten so bildet oder verändert, dass auch mit neu gebildeten oder veränderten programmtechnischen Anweisungen oder Dokumentkomponenten das elektronische Dokument in der für den Benutzer vorgesehenen Form durch die Ausgabeeinheit herstellbar ist.

8. Vorrichtung nach einem der Ansprüche 1 bis 7, dadurch gekennzeichnet, dass die programmtechnische Anweisung so gebildet ist, dass diese in der benutzerseitigen Zugriffseinheit nicht automatisch als Reaktion auf ein Empfangen durch die benutzerseitige Zugriffseinheit zum Ablaufen gestartet wird, sondern ein Ablauf der programmtechnischen Anweisung als Reaktion auf ein vorbestimmtes Ereignis beim Betrieb

der benutzerseitigen Zugriffseinheit, insbesondere einem Anzeigebefehl (onshow, onhide), oder als Reaktion auf das Betätigen eines bevorzugt auf das elektronische Dokument selbst verweisenden Hyperlinks erfolgt.

9. Vorrichtung nach einem der Ansprüche 1 bis 8, dadurch gekennzeichnet, dass die Rekonfigurationseinheit so ausgebildet ist, dass nach einem vorbestimmten Zeitablauf die programmtechnische Anweisung und/oder die Dokumentkomponenten neu gebildet oder verändert werden, auch wenn kein weiterer Zugriff durch den Benutzer erfolgt ist.

10. Vorrichtung nach einem der Ansprüche 1 bis 9, dadurch gekennzeichnet, dass die serverseitige Dekonstruktionseinheit so ausgebildet ist, dass durch das Vorbereiten und Entstrukturieren das elektronische Dokument durch ein Vertauschen, Entfernen, Hinzufügen und/oder Austauschen einzelner inhaltswirksamer Dokumentkomponenten verschlüsselt oder in der Art einer XOR-Funktion verschlüsselt ist.

11. Vorrichtung nach Anspruch 10, dadurch gekennzeichnet, dass die programmtechnische Anweisung als Rekonstruktionsanweisung für das verschlüsselte Dokument wirkt oder einen Zugriffspfad auf eine serverseitig abrufbare Datei beschreibt, die eine Rekonstruktionsanweisung für das verschlüsselte Dokument enthält, und weiter bevorzugt zusätzliche, auf die benutzerseitige Zugriffseinheit und/oder die Ausgabeeinheit wirkende Funktionsbefehle aufweist und/oder selbst verschlüsselt ist.

12. Vorrichtung nach einem der Ansprüche 1 bis 11, dadurch gekennzeichnet, dass die Rekonfigurationseinheit Bestandteil einer der Servereinheit bezogen auf das Datenübertragungsnetz vorgeschalteten Proxyeinheit ist, die zum Speichern und Verwalten von programmtechnischen Anweisungen, Dateibezeichnungen, Pfadbezeichnungen und/oder Darstellungsformbezeichnungen vorgesehen und so ausgebildet ist, dass in der Servereinheit keine Änderung von Dateien und/oder

Dokumentkomponenten als Reaktion auf die Rekonfigurationseinheit erfolgt, wobei bevorzugt die Proxyeinheit zur Ablaufsteuerung einer Benutzersession mit der Servereinheit sowie zur Rechte- und Benutzergruppenverwaltung von durch die Servereinheit angebotenen elektronischen Dokumenten ausgebildet ist.

13. Vorrichtung nach einem der Ansprüche 1 bis 12, dadurch gekennzeichnet, dass der serverseitigen Dekonstruktionseinheit sowie der Rekonfigurationseinheit eine Datenbank (30) zugeordnet ist, die zum Speichern und Zugreifen auf entstrukturierte elektronische Dokumente, zugehörige programmtechnische Anweisungen und/oder Dateinamen von programmtechnischen Anweisungen ausgebildet ist.

14. Verfahren zur geschützten Ausgabe eines elektronischen Dokuments über ein bevorzugt öffentliches Datenübertragungsnetz, insbesondere das Internet, mit den Schritten:

- Zugreifen über das elektronische Datenübertragungsnetz mittels einer benutzerseitigen Zugriffseinheit auf eine das elektronische Dokument zum Ladezugriff anbietende Servereinheit und
- Entfernen und Ausgeben des elektronischen Dokuments mittels einer der benutzerseitigen Zugriffseinheit zugeordneten Ausgabeeinheit in einer für einen Benutzer vorgesehenen Form,
- wobei das empfangene Dokument mindestens eine programmtechnische Anweisung aufweist, die für das Ausgeben durch die Ausgabeeinheit ausführbar ist und eine Bezeichnung einer Datei und/oder eines Pfades im Datenübertragungsnetz und/oder einer Darstellungsform des elektronischen Dokuments enthält, wobei mittels der Bezeichnung Dokumentkomponenten des elektronischen Dokuments geladen und/oder geändert werden,

gekennzeichnet durch die Schritte:

- Vorbereiten und Entstrukturieren des elektronischen Dokuments hinsichtlich der Dokumentkomponenten mittels einer serverseitigen Dekonstruktionseinheit so, dass es

in der für den Benutzer vorgesehenen Form erst nach dem Ausführen einer zugeordneten, durch die Dekonstruktionseinheit entsprechend erzeugten programmtechnischen Anweisung brauchbar ist,

- Bilden oder Verändern der programmtechnischen Anweisung und/oder der Dokumentkomponenten des empfangenen Dokuments durch eine der Servereinheit zugeordneten Rekonfigurations-Einheit so, dass ein erneutes Empfangen des elektronischen Dokuments durch den Benutzer nach einer vorbestimmten, begrenzten Anzahl von weiteren Zugriffen, insbesondere nach nur einem weiteren Zugriff, eine Änderung der Anweisung hinsichtlich einer zugehorigen Dateibezeichnung und/oder einer Pfadbezeichnung und/oder einer Darstellungsformbezeichnung, und/oder eine Änderung der damit zu ladenden Dokumentkomponenten, gegenüber einem vorhergehenden Empfangen und Ausgeben, bewirkt.

Fig. 1

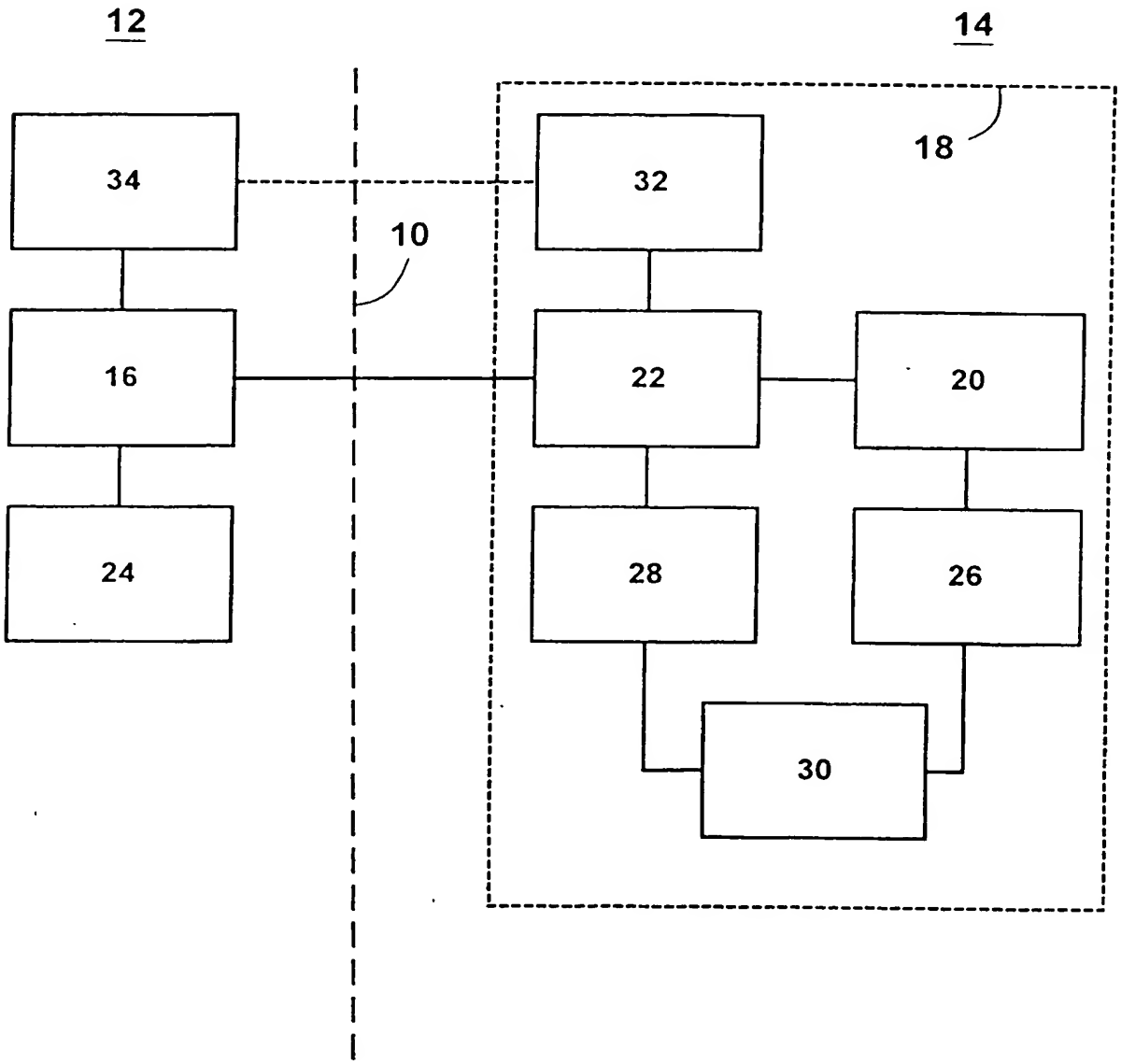
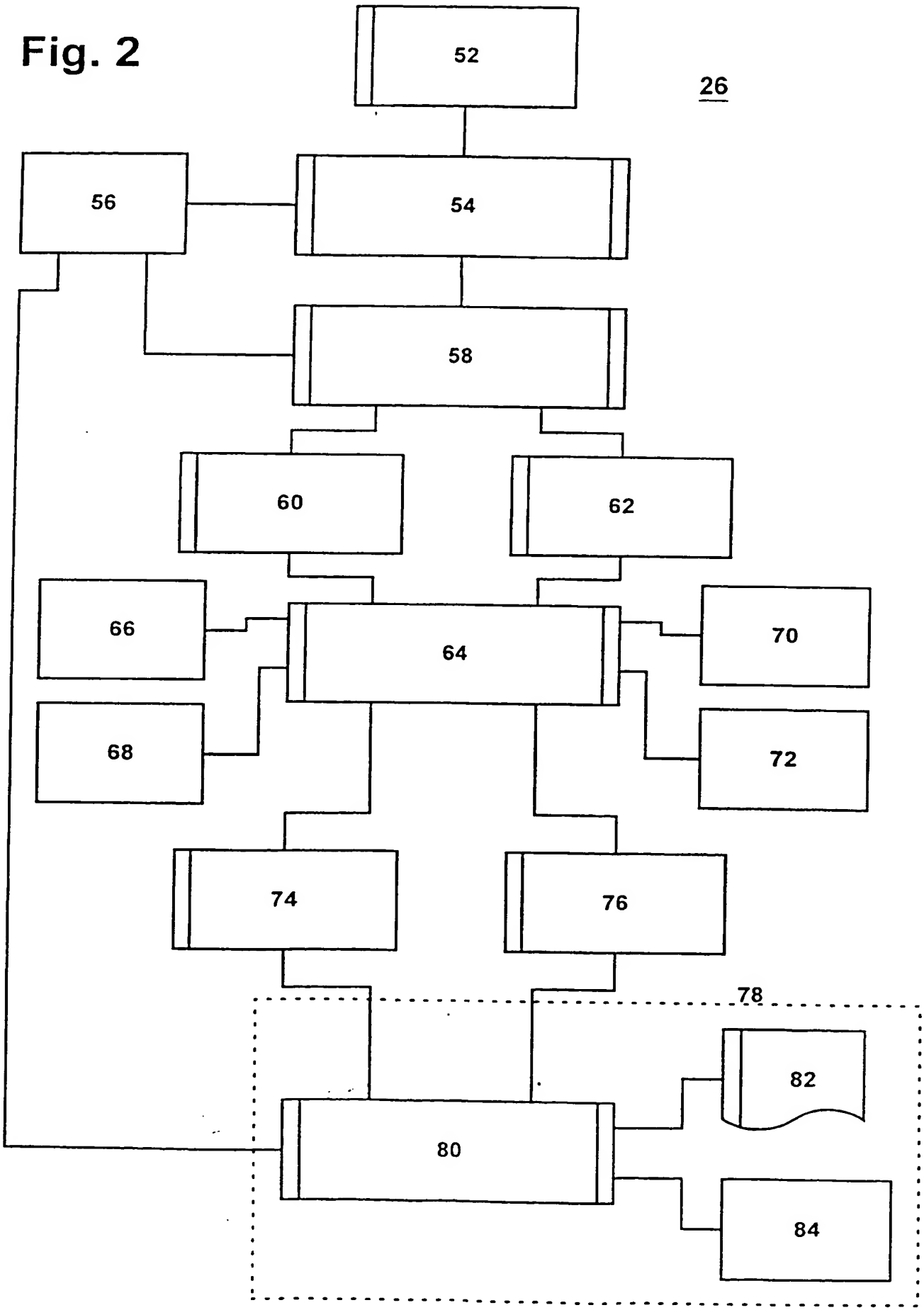


Fig. 2



INTERNATIONAL SEARCH REPORT

In. ational Application No
PCT/EP 00/10750

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F1/00

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, WPI Data, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	GB 2 325 767 A (NIPPON TELEGRAPH & TELEPHONE) 2 December 1998 (1998-12-02) abstract; figure 1 page 8, line 10 -page 14, line 19	1,14
A	WO 98 44402 A (BRAMHILL IAN DUNCAN ;SIMS MATTHEW ROBERT CHARLES (GB); BRITISH TEL) 8 October 1998 (1998-10-08) page 9, line 15 -page 15, line 12	1-4,7,8, 14

☐ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
5 March 2001	12/03/2001

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Sigolo, A

INTERNATIONAL SEARCH REPORT

Information on patent family members

In ational Application No
PCT/EP 00/10750

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
GB 2325767 A	02-12-1998	JP 11242625 A	07-09-1999
		US 6014696 A	11-01-2000
WO 9844402 A	08-10-1998	AU 6414098 A	22-10-1998
		EP 0970411 A	12-01-2000

INTERNATIONALER RECHERCHENBERICHT

In. .ationales Aktenzeichen
PCT/EP 00/10750

A. KLASSIFIZIERUNG DES ANMELDUNGSGEGENSTANDES
IPK 7 G06F1/00

Nach der internationalen Patentklassifikation (IPK) oder nach der nationalen Klassifikation und der IPK

B. RECHERCHIERTE GEBIETE

Recherchierter Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)
IPK 7 G06F H04L

Recherchierte aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)
EPO-Internal, WPI Data, PAJ

C. ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
A	GB 2 325 767 A (NIPPON TELEGRAPH & TELEPHONE) 2. Dezember 1998 (1998-12-02) Zusammenfassung; Abbildung 1 Seite 8, Zeile 10 -Seite 14, Zeile 19 ---	1,14
A	WO 98 44402 A (BRAMHILL IAN DUNCAN ;SIMS MATTHEW ROBERT CHARLES (GB); BRITISH TEL) 8. Oktober 1998 (1998-10-08) Seite 9, Zeile 15 -Seite 15, Zeile 12 -----	1-4,7,8, 14

☐ Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen

☒ Siehe Anhang Patentfamilie

* Besondere Kategorien von angegebenen Veröffentlichungen :
"A" Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist
"E" älteres Dokument, das jedoch erst am oder nach dem internationalen Anmeldedatum veröffentlicht worden ist
"L" Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)
"O" Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht
"P" Veröffentlichung, die vor dem internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist
"T" Spätere Veröffentlichung, die nach dem internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist
"X" Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erfinderischer Tätigkeit beruhend betrachtet werden
"Y" Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann naheliegend ist
"&" Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der internationalen Recherche	Absendedatum des internationalen Recherchenberichts
5. März 2001	12/03/2001
Name und Postanschrift der Internationalen Recherchenbehörde Europäisches Patentamt, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Bevollmächtigter Bediensteter Sigolo, A

INTERNATIONALER RECHERCHENBERICHT

Angaben zu Veröffentlichungen, die zur selben Patentfamilie gehören

In..ationales Aktenzeichen
PCT/EP 00/10750

Im Recherchenbericht angeführtes Patentdokument	Datum der Veröffentlichung	Mitglied(er) der Patentfamilie	Datum der Veröffentlichung
GB 2325767 A	02-12-1998	JP 11242625 A	07-09-1999
		US 6014696 A	11-01-2000
WO 9844402 A	08-10-1998	AU 6414098 A	22-10-1998
		EP 0970411 A	12-01-2000